



Examen de muestra

Edición 201807

Copyright © EXIN Holding B.V. 2018. All rights reserved.
EXIN® is a registered trademark.

No part of this publication may be published, reproduced, copied or stored in a data processing system or circulated in any form by print, photo print, microfilm or any other means without written permission by EXIN.



Índice

Introducción	4
Examen de muestra	5
Soluciones	16
Evaluación	40

Introducción

Éste es el examen de muestra de EXIN DevOps Professional (DEVOPSP.SP). Las normas y reglamentos de exámenes EXIN se aplican a este examen.

Este examen consiste en 40 preguntas del tipo opciones múltiples. Cada pregunta tiene un número de respuestas posibles, de las cuales sólo una es correcta.

El número máximo de puntos que se pueden obtener en este examen es de 40. Cada respuesta correcta tiene un valor de un punto. Si usted consigue 26 puntos o más, habrá aprobado el examen.

El tiempo permitido para este examen es de 90 minutos.

¡Buena suerte!

Examen de muestra

1 / 40

¿Cuál es un principio del Manifiesto Ágil?

- A) Crear una cultura de aprendizaje continuo y dinámico
- B) Crear bucles de retroalimentación (*feedback*) y de control predictivo (*feedforward*) en nuestro sistema de trabajo.
- C) Entregar software funcional frecuentemente, desde un par de semanas hasta un par de meses
- D) Aumentar el flujo dando visibilidad al trabajo, reduciendo el tamaño de los lotes y los intervalos de trabajo, y creando calidad

2 / 40

¿A cuál de los tres caminos (*Three Ways*) corresponde «institucionalizar la mejora del trabajo diario»?

- A) Aprendizaje y experimentación continuos
- B) Retroalimentación
- C) Flujo

3 / 40

¿Cuál es la diferencia entre un Sistema de contratación (SoE) y un Sistema de registro (SoR) en cuanto al ritmo de cambio?

- A) Los SoE y SoR normalmente tienen el mismo ritmo de cambio.
- B) Un SoE normalmente tiene un ritmo de cambio mucho más elevado que un SoR.
- C) Un SoE normalmente tiene un ritmo de cambio mucho más bajo que un SoR.
- D) Esta relación varía según el sistema de información.

4 / 40

¿Cuál es la ventaja de que Desarrollo y Operaciones utilicen una herramienta compartida?

- A) Un backlog unificado, en el que todos dan prioridad a los proyectos de mejora desde una perspectiva global.
- B) Los desarrolladores reciben información sobre cómo funcionan sus aplicaciones en producción, lo que incluye arreglarlas cuando fallen.
- C) Permite que el equipo realice despliegues durante el horario normal de trabajo y que lleve a cabo cambios sencillos.
- D) Transforma los conocimientos de Operaciones en código automatizado que pueda ser mucho más fiable y ampliamente reutilizado.

5 / 40

Se pueden crear resultados más orientados al mercado mediante una mejor integración de las competencias de Operaciones en los equipos de Desarrollo, haciendo que ambos sean más eficientes y productivos.

¿Cuál sería la **mejor** estrategia para conseguirlo?

- A) Asignar un enlace de Desarrollo al equipo de Operaciones
- B) Crear infraestructuras de autoservicio
- C) Externalizar el equipo de Operaciones
- D) Formar a los desarrolladores para que hagan el trabajo de Operaciones

6 / 40

Un banco necesita plazos más largos para introducir ofertas nuevas o modificadas en el mercado, debido a los retrasos en la creación de nuevos entornos por parte de los ingenieros de Operaciones.

¿Qué hay de **cierto** en cuanto a la creación automática de entornos?

- A) Los entornos creados automáticamente se pueden utilizar en todos los entornos.
- B) Los entornos creados automáticamente se pueden utilizar en todos los entornos, excepto en el entorno de producción debido a las restricciones por motivos de seguridad.
- C) DevOps requiere una revisión por parte de Operaciones cuando se crean automáticamente entornos en producción, de acuerdo con el principio de que cuatro ojos ven mejor que dos.
- D) DevOps requiere el consentimiento manual de Operaciones cuando se crean automáticamente entornos en producción, de acuerdo con el principio de que cuatro ojos ven mejor que dos.

7 / 40

¿Qué valor debería ser incluido en un bloque de procesos de un mapa de valor?

- A) Telemetría de aplicaciones
- B) Porcentaje de completitud y exactitud
- C) Velocidad del equipo
- D) WiP (Trabajo en curso)

8 / 40

Para minimizar los riesgos para el negocio cuando se trabaja con DevOps, ¿cuál es el objetivo **principal** del control de versiones?

- A) Garantizar la capacidad de alertar cuando la configuración cambie con respecto al estado deseado
- B) Garantizar la capacidad de volver a crear el estado anterior del entorno de pruebas
- C) Garantizar la capacidad de volver a crear el entorno de producción y ensamblar procesos
- D) Garantizar la capacidad de compartir el código fuente entre distintos equipos de desarrolladores

9 / 40

Un equipo de TI se reúne para revisar algunos cambios que deberían hacerse para ir más allá en su adopción de DevOps. Deben acordar una Definición de Terminado (DoD) que esté en consonancia con los principios de DevOps.

¿Qué Definición de Terminado es adecuada para DevOps?

- A) El código ha sido integrado en la rama principal y ha superado las pruebas unitarias automatizadas.
- B) El código está funcionando según lo previsto en el portátil del desarrollador y ha superado las pruebas unitarias.
- C) El código está funcionando en un entorno similar al de producción y ha superado las pruebas de aceptación del usuario.

10 / 40

¿Qué herramientas pueden **mejor** utilizarse para automatizar la construcción y configuración de entornos?

- A) Un sistema de tickets para la provisión de un entorno de desarrollo, prueba o aceptación
- B) Una herramienta que copie la configuración del entorno de producción a los entornos de desarrollo, prueba y aceptación
- C) Ficheros de configuración por entorno que se distribuyan y se mantengan de forma manual para sincronizar los entornos
- D) Herramientas de gestión de configuración de la infraestructura programable que permitan a los mismos programadores de cambiar los entornos

11 / 40

Un objetivo específico de diseño de un paquete de pruebas automatizadas es encontrar errores durante las pruebas lo más pronto posible. La pirámide ideal de automatización de pruebas muestra el orden correcto en el que deben realizarse las pruebas.

¿Qué prueba deberá realizarse **primero**?

- A) Prueba de APIs automatizada
- B) Prueba de componentes automatizada
- C) Prueba de integración automatizada
- D) Prueba unitaria automatizada

12 / 40

Un equipo de DevOps quiere aumentar la velocidad mediante la utilización de un desarrollo orientado a las pruebas.

¿Cuál es el orden correcto de las acciones?

- A) A) Refactorizar
B) Escribir un caso de prueba
C) Escribir el código funcional
- B) A) Escribir el código funcional
B) Escribir un caso de prueba
C) Refactorizar
- C) A) Escribir un caso de prueba
B) Escribir el código funcional
C) Refactorizar

13 / 40

Una aseguradora contrata a un profesional de DevOps que asesore al equipo de DevOps para elegir una estrategia de ramificación. El profesional de DevOps evalúa dos estrategias:

1. Optimizar para la productividad individual
2. Optimizar para la productividad del equipo

¿Qué afirmación sobre estas dos estrategias es **verdadera**?

- A) Ambas estrategias dan lugar a la misma cantidad de esfuerzo para fusionar.
- B) La estrategia 1 da lugar a mucho más esfuerzo para fusionar el código que la 2.
- C) La estrategia 2 da lugar a mucho más esfuerzo para fusionar el código que la 1.

14 / 40

Una empresa minorista ha cambiado radicalmente el proceso de desarrollo desde un enfoque en cascada a un enfoque DevOps. Se deben tomar muchas decisiones en poco tiempo. Esto está causando una cierta deuda técnica. Por ejemplo, el tiempo de comercialización ha aumentado drásticamente. No obstante, existen maneras de eliminar muchos de los retrasos.

El mapeo de la cadena de valor muestra que el 20 % del tiempo del sprint se dedica a portar código, todo mantenido en ramas de código independientes.

¿Qué solución para este tipo de deuda técnica dará lugar a un flujo más rápido?

- A) Adoptar un desarrollo basado en rama principal
- B) Reproducir las pruebas falladas en las estaciones de trabajo del desarrollador
- C) Empezar a hacer *canary releases* (técnica de lanzamiento de bajo riesgo)
- D) Usar más telemetría

15 / 40

¿Cuál es una característica de una arquitectura que permite productividad, testabilidad y seguridad?

- A) Tener interfaces débilmente definidas
- B) Ser fuertemente acoplada
- C) Tener APIs bien definidas

16 / 40

¿Cuál es la **mejor** estrategia de telemetría para resolver problemas cuando se trabaja con DevOps?

- A) Invertir en herramientas de telemetría es importante, y debería centrarse en el entorno de producción, *pipeline* de despliegue y preproducción.
- B) Invertir en herramientas de telemetría es importante, y debería centrarse solo en el entorno de producción.
- C) Invertir en herramientas de telemetría no es importante; la atención debería centrarse en reiniciar los servicios redundantes, lo cual es mucho más barato.
- D) Invertir en herramientas de telemetría no es importante; la atención debería centrarse en el usuario de la aplicación, que es una fuente de información de monitorización mucho más completa y barata.

17 / 40

En la infraestructura de monitorización, los datos se deberían recopilar a partir de tres capas.

¿Cuál **no** es una de esas tres capas?

- A) Aplicación
- B) Lógica de negocio
- C) Métricas de negocio
- D) Sistema operativo

18 / 40

¿Para qué tipo de destinatarios aporta valor el acceso de autoservicio a la telemetría?

- A) Sólo desarrolladores
- B) Sólo desarrolladores y operadores
- C) Sólo desarrolladores, operadores y partes interesadas
- D) Desarrolladores, operadores, partes interesadas y clientes

19 / 40

En un entorno que tiene pruebas automatizadas, procesos de despliegue rápido y telemetría adecuada, ¿qué técnica ofrece **mayores** beneficios de DevOps al negocio?

- A) Hacer una corrección preventiva (*fix forward*)
- B) Restaurar la última versión (*roll back*)
- C) Retirar los servidores averiados
- D) Desactivar las funcionalidades que fallan

20 / 40

Una empresa de software ha redactado los siguientes requisitos para la guía de lanzamientos:

- Recuento y gravedad de defectos: ¿La aplicación funciona conforme al diseño?
- Cobertura de monitorización: ¿La cobertura de monitorización es suficiente para restaurar el servicio cuando las cosas no vayan bien?
- Arquitectura de sistemas: ¿Está el servicio lo suficientemente acoplado como para resolver una tasa elevada de incidentes en producción?
- Tipo/frecuencia de mensajes de alerta a buscapersonas: ¿La aplicación está generando un número insostenible de alertas en producción?

¿Qué requisito de la guía de lanzamientos **no** cumple con la forma de trabajar de DevOps?

- A) Recuento y gravedad de defectos: ¿La aplicación funciona realmente conforme al diseño?
- B) Cobertura de monitorización: ¿La cobertura de monitorización es suficiente para restaurar el servicio cuando las cosas no vayan bien?
- C) Arquitectura de sistemas: ¿Está el servicio lo suficientemente acoplado como para resolver una tasa elevada de incidentes en producción?
- D) Tipo/frecuencia de mensajes de alerta a buscapersonas: ¿La aplicación está generando un número insostenible de alertas en producción?

21 / 40

¿Cuál es una característica única de la revisión de preparación para el traspaso (HRR) y **no** de la revisión de preparación para el lanzamiento (LRR)?

- A) La HRR es mucho más estricta y tiene estándares de aceptación más elevados.
- B) La HRR es autoevaluada por los equipos de producto.
- C) La HRR se aprueba antes de que cualquier nuevo servicio se ponga a disposición del público.
- D) La HRR se aprueba antes de que cualquier nuevo servicio reciba tráfico de producción activo.

22 / 40

Una de las técnicas más potentes en el diseño de interacción y experiencia de usuario (UX) es la investigación contextual.

¿Cuál es la **mejor** descripción de investigación contextual?

- A) El equipo de producto pide a los usuarios que respondan a una demostración de la aplicación por parte del equipo de producto.
- B) El equipo de producto entrevista a los usuarios sobre el uso de la aplicación en sus hogares.
- C) El equipo de producto observa a los usuarios que utilizan la aplicación en su entorno natural.
- D) El equipo de producto estudia a los usuarios durante la prueba de aceptación de usuario en una sala especial con equipamientos de prueba.

23 / 40

La startup ABC está teniendo serios problemas en el desarrollo de las funcionalidades para una aplicación de navegación móvil que satisfaga las expectativas del usuario.

¿Cuál es la **mejor** estrategia para garantizar que las funcionalidades estén directamente alineadas con las expectativas del usuario respecto a la experiencia de usuario de la aplicación de navegación móvil ?

- A) Desarrollar más plenamente la experiencia de usuario de la aplicación de navegación móvil, de manera que los consumidores dispongan de funcionalidades y opciones más avanzadas para navegar en la experiencia de usuario, ofreciéndoles más opciones para personalizar la misma.
- B) Desarrollar la aplicación de navegación móvil basada en la experiencia de usuario, en la que se seleccionan aleatoriamente a consumidores para mostrarles una de dos versiones de una experiencia de usuario entre las que pueden elegir, o bien la versión de referencia (control) o una variante.
- C) Efectuar correcciones sobre las características actuales relacionadas con la experiencia de usuario de la aplicación de navegación móvil, de manera que ABC se ciña a la funcionalidad básica de la experiencia de usuario sin sacrificar la calidad.
- D) Lanzar al mercado una funcionalidad de la aplicación de navegación móvil para recibir la opinión de los consumidores sobre la funcionalidad actual durante un período de tres semanas.

24 / 40

¿Qué estrategia puede utilizar un equipo de desarrollo para ayudar a conseguir los resultados de negocio esperados?

- A) Ejecución con esmero numerosos experimentos
- B) Pruebas de regresión completas
- C) Desarrollo guiado por hipótesis
- D) SDLC (ciclo de vida de desarrollo de software)

25 / 40

La empresa ABC ha tenido dificultades para comprender el impacto de los cambios en el código en su aplicación de nóminas. Actualmente, los cambios implementados en el código están mal documentados. Por ejemplo, la última corrección de código se documentó como se indica a continuación: «Corrección problema n.º 1801 para la aplicación de nóminas». La empresa se da cuenta de que ésta es una *pull request* mala.

Una buena *pull request* incluye lo siguiente:

- ¿Por qué implementamos la corrección?
- ¿Quién implementa la corrección?

¿Qué más se **debe** incluir en una buena *pull request*?

- A) Las unidades de negocio afectadas por la corrección
- B) Los posibles riesgos y contramedidas relacionados con la implementación de la corrección
- C) El plan de apoyo al lanzamiento para implementar la corrección

26 / 40

Considere el siguiente escenario:

«El desarrollador revisa el código con un compañero que le da su opinión al respecto.».

¿Qué técnica de revisión se describe aquí?

- A) Revisión supervisada por un par (*over-the-shoulder*)
- B) Programación en parejas
- C) Revisión por pares
- D) Revisión asistida por herramientas

27 / 40

¿Qué técnica de revisión resulta directamente en una disminución de los errores de codificación?

- A) Revisión supervisada por un par (*over-the-shoulder*)
- B) Programación en parejas
- C) Revisión por pares
- D) Revisión asistida por herramientas

28 / 40

Un proveedor de servicios en la nube quiere aumentar la capacidad de su servicio utilizando un mono del ejército de los simios (*The Simian Army*).

¿Qué mono es necesario en este caso?

- A) Mono Doctor (*Doctor Monkey*)
- B) Mono Conserje (*Janitor Monkey*)
- C) Mono Latencia (*Latency Monkey*)

29 / 40

La empresa ABC está adoptando DevOps en su manera de trabajar y quiere fomentar un entorno de aprendizaje que sea abierto y donde no se culpabilice a nadie. ABC sufrió recientemente un fallo importante en una aplicación y pudo restaurar el servicio.

¿Cuál es la **primera** tarea que debe completarse durante la reunión de análisis *a posteriori* libre de culpa?

- A) Crear un cronograma de eventos relevantes tal y como ocurrieron durante el fallo de la aplicación
- B) Identificar contramedidas para evitar que el fallo de la aplicación vuelva a ocurrir en el futuro
- C) Identificar la causa raíz del fallo de la aplicación para proponer una medida correctiva con el fin de evitar que el mismo fallo vuelva a ocurrir en el futuro
- D) Publicar los resultados del análisis *a posteriori* en una ubicación centralizada de manera que toda la organización pueda acceder a ellos y aprender del fallo de la aplicación

30 / 40

¿Qué es necesario cuando se crea resiliencia inyectando fallos en producción?

- A) Definir el modo de fallo
- B) Organizar reuniones de análisis *a posteriori*
- C) Formar al personal
- D) Usar un entorno de prueba

31 / 40

¿Cuál es el **primer** paso de un plan para utilizar gamificación para simular accidentes (*Game Days*)?

- A) Definir y ejecutar simulacros
- B) Identificar y abordar problemas, y probar
- C) Planificar la interrupción del servicio
- D) Preparar y eliminar los puntos únicos de fallo

32 / 40

¿Cuál es un ejemplo de un requisito no funcional?

- A) Tener compatibilidad con versiones anteriores y posteriores
- B) Permitir informar sobre retrasos en los planes
- C) Registrar transacciones financieras para un sistema de reservas de hotel

33 / 40

¿Qué actividad se debería realizar para la creación satisfactoria de historias de usuario de operaciones reutilizables?

- A) Asociar las historias de usuario de operaciones a las mejoras y defectos de desarrollo pertinentes
- B) Definir actividades dentro del proceso de traspaso y posteriormente automatizar dichas actividades mediante el uso de herramientas y flujos de trabajo de apoyo adecuados
- C) Identificar todas las actividades de trabajo operativo necesarias y los actores necesarios para completarlas

34 / 40

Considere los siguientes elementos:

1. Solicitudes de cambio
2. Herramientas del *pipeline* de despliegue
3. Ejecutables de programas compilados
4. Tutoriales y estándares

¿Qué **dos** elementos se suelen guardar en un repositorio único de código fuente compartido?

- A) 1 y 2
- B) 1 y 4
- C) 2 y 3
- D) 2 y 4

35 / 40

¿Cuál es el objetivo de convertir descubrimientos locales en mejoras globales?

- A) Elevar el nivel de las prácticas vigentes, no solo de Desarrollo y Operaciones, sino de toda la organización
- B) Hacer que el aprovechamiento del conocimiento colectivo sea más sencillo, tanto para los servicios nuevos como para los ya existentes
- C) Hacer que la cultura de trabajo sea más colaborativa, y que los sistemas sean más seguros y resilientes
- D) Reforzar una cultura en la que todos se sientan cómodos y responsables

36 / 40

Los desarrolladores pueden facilitar que cualquier ingeniero cree y utilice de forma correcta las normas de registro y de criptografía en sus aplicaciones y entornos.

¿Qué opción **no** es un elemento del repositorio de código fuente compartido compatible con esto?

- A) Librerías de código y sus configuraciones recomendadas
- B) Paquetes de despliegue
- C) Paquetes y compilaciones de sistemas operativos (SO)
- D) Herramientas de gestión de preguntas de seguridad

37 / 40

Cuando los desarrolladores introducen código, siempre existe el riesgo de permitir un acceso no autorizado.

¿Qué control **no** mitiga ese riesgo?

- A) Revisiones de código
- B) Pruebas de código
- C) Parches efectivos
- D) Pruebas de intrusión

38 / 40

¿Cuál es un ejemplo de creación de telemetría para una aplicación?

- A) Cambios en el sistema operativo (SO)
- B) Revisión diaria de los registros de sistema
- C) Cambios en los grupos de seguridad
- D) Restablecimientos de las contraseñas de los usuarios

39 / 40

¿Qué proceso sirve como control primario para reducir los riesgos de seguridad y de Operaciones, y también soporta los requisitos de conformidad?

- A) Proceso de gestión de cambios
- B) Proceso de gestión de la configuración
- C) Proceso de gestión de entregas y despliegues
- D) Proceso de gestión del nivel de servicio

40 / 40

¿Qué supone un inconveniente a la hora de adoptar la segregación de funciones como medida de control?

- A) La segregación de funciones como medida de control puede dificultar a menudo los esfuerzos de desarrollo al ralentizar y reducir la retroalimentación que reciben los ingenieros sobre su trabajo.
- B) La segregación de funciones como medida de control requiere que el desarrollador envíe los cambios al responsable de la librería de código, quien revisará y aprobará el cambio antes de que éste pase a producción.
- C) La segregación de funciones como medida de control da lugar a trabajo innecesario, ya que requiere la inspección del código que se sube, y las revisiones de código, que proporcionan la confianza necesaria sobre la calidad de nuestro trabajo.

Soluciones

1 / 40

¿Cuál es un principio del Manifiesto Ágil?

- A) Crear una cultura de aprendizaje continuo y dinámico
- B) Crear bucles de retroalimentación (*feedback*) y de control predictivo (*feedforward*) en nuestro sistema de trabajo.
- C) Entregar software funcional frecuentemente, desde un par de semanas hasta un par de meses
- D) Aumentar el flujo dando visibilidad al trabajo, reduciendo el tamaño de los lotes y los intervalos de trabajo, y creando calidad

- A) Incorrecto. Esto corresponde al principio de aprendizaje y experimentación continuos.
- B) Incorrecto. Esto corresponde al principio de retroalimentación.
- C) Correcto. Este es uno de los principios clave del Manifiesto Ágil. Otros principios son la necesidad de equipos pequeños, con iniciativa y que trabajen con un modelo de gestión de plena confianza. (Bibliografía: A, Introducción de la Parte 1)
- D) Incorrecto. Esto corresponde al principio del flujo.

2 / 40

¿A cuál de los tres caminos (*Three Ways*) corresponde «institucionalizar la mejora del trabajo diario»?

- A) Aprendizaje y experimentación continuos
- B) Retroalimentación
- C) Flujo

- A) Correcto. El principio del tercer camino requiere «institucionalizar la mejora del trabajo diario», convirtiendo los conocimientos locales en conocimientos globales que pueda utilizar toda la organización, así como inyectar continuamente tensión en nuestro trabajo diario. (Bibliografía: A, Capítulo 4)
- B) Incorrecto. «Institucionalizar la mejora del trabajo diario» es un principio de aprendizaje y experimentación continuos.
- C) Incorrecto. «Institucionalizar la mejora del trabajo diario» es un principio de aprendizaje y experimentación continuos.

3 / 40

¿Cuál es la diferencia entre un Sistema de contratación (SoE) y un Sistema de registro (SoR) en cuanto al ritmo de cambio?

- A) Los SoE y SoR normalmente tienen el mismo ritmo de cambio.
 - B) Un SoE normalmente tiene un ritmo de cambio mucho más elevado que un SoR.
 - C) Un SoE normalmente tiene un ritmo de cambio mucho más bajo que un SoR.
 - D) Esta relación varía según el sistema de información.
-
- A) Incorrecto. Un SoE normalmente tiene un ritmo de cambio mucho más elevado que un SoR.
 - B) Correcto. El ritmo de cambio de un SoE es más elevado ya que representa la interfaz de usuario. Además, un SoR es más complejo de cambiar. (Bibliografía: A, Capítulo 5)
 - C) Incorrecto. Un SoE normalmente tiene un ritmo de cambio mucho más elevado que un SoR.
 - D) Incorrecto. Se puede hacer una afirmación generalizada. Un SoE normalmente tiene un ritmo de cambio mucho más elevado que un SoR.

4 / 40

¿Cuál es la ventaja de que Desarrollo y Operaciones utilicen una herramienta compartida?

- A) Un backlog unificado, en el que todos dan prioridad a los proyectos de mejora desde una perspectiva global.
 - B) Los desarrolladores reciben información sobre cómo funcionan sus aplicaciones en producción, lo que incluye arreglarlas cuando fallen.
 - C) Permite que el equipo realice despliegues durante el horario normal de trabajo y que lleve a cabo cambios sencillos.
 - D) Transforma los conocimientos de Operaciones en código automatizado que pueda ser mucho más fiable y ampliamente reutilizado.
-
- A) Correcto. Con un backlog unificado todos dan prioridad a los proyectos de mejora desde una perspectiva global, seleccionando el trabajo que tenga el valor más elevado para la organización o que tenga el mayor impacto en la reducción de la deuda técnica. (Bibliografía: A, Capítulo 6)
 - B) Incorrecto. Esta es una ventaja que Desarrollo comparte con Operaciones cuando se hacen rotaciones en los turnos de guardia.
 - C) Incorrecto. Esta es una ventaja de automatizar y permitir los lanzamientos de bajo riesgo.
 - D) Incorrecto. Esta es una ventaja de integrar a los ingenieros de Operaciones en nuestros equipos de servicio.

5 / 40

Se pueden crear resultados más orientados al mercado mediante una mejor integración de las competencias de Operaciones en los equipos de Desarrollo, haciendo que ambos sean más eficientes y productivos.

¿Cuál sería la **mejor** estrategia para conseguirlo?

- A) Asignar un enlace de Desarrollo al equipo de Operaciones
- B) Crear infraestructuras de autoservicio
- C) Externalizar el equipo de Operaciones
- D) Formar a los desarrolladores para que hagan el trabajo de Operaciones

- A) Incorrecto. Debería ser a la inversa: «Asignar un enlace de Operaciones al equipo de Desarrollo».
- B) Correcto. Se trata de una de las tres estrategias generales para integrar Operaciones en Desarrollo:
 - Crear capacidades de autoservicio para permitir que los desarrolladores de los equipos de servicio sean productivos.
 - Integrar a los ingenieros de Operaciones en los equipos de servicios.
 - Asignar enlaces de Operaciones a los equipos de servicios cuando no sea posible la integración de Operaciones. (Bibliografía: A, Capítulo 8)
- C) Incorrecto. No se recomienda la externalización como estrategia. Los empleados del subcontratista estarán todavía más alejados del equipo de Desarrollo.
- D) Incorrecto. Los desarrolladores pueden utilizar la infraestructura programable; no obstante, esto no hará que la función de Operaciones sea obsoleta.

6 / 40

Un banco necesita plazos más largos para introducir ofertas nuevas o modificadas en el mercado, debido a los retrasos en la creación de nuevos entornos por parte de los ingenieros de Operaciones.

¿Qué hay de **cierto** en cuanto a la creación automática de entornos?

- A) Los entornos creados automáticamente se pueden utilizar en todos los entornos.
 - B) Los entornos creados automáticamente se pueden utilizar en todos los entornos, excepto en el entorno de producción debido a las restricciones por motivos de seguridad.
 - C) DevOps requiere una revisión por parte de Operaciones cuando se crean automáticamente entornos en producción, de acuerdo con el principio de que cuatro ojos ven mejor que dos.
 - D) DevOps requiere el consentimiento manual de Operaciones cuando se crean automáticamente entornos en producción, de acuerdo con el principio de que cuatro ojos ven mejor que dos.
-
- A) Correcto. El despliegue continuo no requiere que se realicen acciones manuales mediante el uso de *scripts* (secuencias de comandos). Estos *scripts* deben automatizarse. Esto es válido para todos los entornos. (Bibliografía: A, Capítulo 9)
 - B) Incorrecto. El entorno de producción debería crearse y mantenerse de forma totalmente automática.
 - C) Incorrecto. No es necesario revisar los *scripts* automatizados. Los procesos automatizados se monitorizan en todo caso.
 - D) Incorrecto. No es necesario aplicar el principio de que cuatro ojos ven mejor que dos, ya que la aceptación de los *scripts* y la herramienta de despliegue de los *scripts* están bajo el control de cambios.

7 / 40

¿Qué valor debería ser incluido en un bloque de procesos de un mapa de valor?

- A) Telemetría de aplicaciones
- B) Porcentaje de completitud y exactitud
- C) Velocidad del equipo
- D) WiP (Trabajo en curso)

- A) Incorrecto. La telemetría de aplicaciones se utiliza para monitorizar el comportamiento de una aplicación mientras está en funcionamiento para informar sobre su estado, proporcionando una retroalimentación rápida y precisa, así como la detección de errores.
- B) Correcto. Esto debería incluirse en un bloque de procesos del mapa de valor ya que es parte de los indicadores clave para entender la calidad ofrecida por esa cadena de valor según los requisitos del negocio. (Bibliografía: A, Capítulo 6)
- C) Incorrecto. La velocidad del equipo es un valor que se mide durante un sprint y se utiliza durante una reunión del sprint para calcular la cantidad de trabajo que se puede asumir durante la iteración.
- D) Incorrecto. WiP es un concepto que debe incluirse o derivar de un panel Kanban y no forma parte del mapa de valor ni proporciona ningún valor para el análisis de la cadena de valor.

8 / 40

Para minimizar los riesgos para el negocio cuando se trabaja con DevOps, ¿cuál es el objetivo **principal** del control de versiones?

- A) Garantizar la capacidad de alertar cuando la configuración cambie con respecto al estado deseado
 - B) Garantizar la capacidad de volver a crear el estado anterior del entorno de pruebas
 - C) Garantizar la capacidad de volver a crear el entorno de producción y ensamblar procesos
 - D) Garantizar la capacidad de compartir el código fuente entre distintos equipos de desarrolladores
-
- A) Incorrecto. Aunque el control de versiones es fundamental para que otras herramientas utilicen su información para detectar alteraciones en la configuración con respecto al estado deseado, este no es el objetivo del propio control de versiones, ni la mejor respuesta en este caso, ya que el control de versiones debería permitir una replicación más rápida de los entornos para mantener la calidad del servicio y del flujo.
 - B) Incorrecto. Esto podría ser una ventaja del control de versiones, pero no es el objetivo principal del mismo cuando se trabaja con DevOps. Además, el entorno de pruebas se suele volver a generar para cada prueba, así que devolverlo a un estado anterior no resultaría muy útil.
 - C) Correcto. El control de versiones en DevOps debería permitir volver a crear entornos de producción, así como compilar procesos. Las organizaciones que adoptan este enfoque funcionan mejor que las que no lo hacen. (Bibliografía: A, Capítulo 9)
 - D) Incorrecto. Esta es la finalidad del control de versiones cuando se utiliza solo con un enfoque de Desarrollo, pero no es ese el objetivo cuando se trabaja con DevOps, ya que el control de versiones en ámbito DevOps requiere que todas las áreas guarden sus artefactos, herramientas y procesos en él para mejorar el rendimiento, el flujo, la comunicación entre equipos y la previsibilidad, lo que lleva a una reducción de las ineficiencias en todos los entornos.

9 / 40

Un equipo de TI se reúne para revisar algunos cambios que deberían hacerse para ir más allá en su adopción de DevOps. Deben acordar una Definición de Terminado (DoD) que esté en consonancia con los principios de DevOps.

¿Qué Definición de Terminado es adecuada para DevOps?

- A) El código ha sido integrado en la rama principal y ha superado las pruebas unitarias automatizadas.
 - B) El código está funcionando según lo previsto en el portátil del desarrollador y ha superado las pruebas unitarias.
 - C) El código está funcionando en un entorno similar al de producción y ha superado las pruebas de aceptación del usuario.
-
- A) Incorrecto. Esta es la Definición de Terminado estándar desde una perspectiva de desarrollador y no logra el objetivo de DevOps, que consiste en aportar valor a la empresa, por lo que la Definición de Terminado debe incluir el envío del código de trabajo a un entorno semejante al de producción, y no la integración en la rama principal.
 - B) Incorrecto. El hecho de que todo funcione en el portátil del desarrollador no proporciona ninguna garantía de que funcionará en el entorno similar al de producción.
 - C) Correcto. Esto garantiza que el código está funcionando según lo esperado y es potencialmente desplegable. (Bibliografía: A, Capítulo 9)

10 / 40

¿Qué herramientas pueden **mejor** utilizarse para automatizar la construcción y configuración de entornos?

- A) Un sistema de tickets para la provisión de un entorno de desarrollo, prueba o aceptación
 - B) Una herramienta que copie la configuración del entorno de producción a los entornos de desarrollo, prueba y aceptación
 - C) Ficheros de configuración por entorno que se distribuyan y se mantengan de forma manual para sincronizar los entornos
 - D) Herramientas de gestión de configuración de la infraestructura programable que permitan a los mismos programadores de cambiar los entornos
-
- A) Incorrecto. Un sistema de tickets para el suministro de entornos dificultaría el flujo en el *pipeline* de despliegue, generaría más traspasos y, por lo tanto, equivaldría a desperdicio. Se deberían utilizar herramientas para automatizar el suministro de entornos.
 - B) Incorrecto. Esta no es la forma de trabajar propuesta en ámbito DevOps. Se debe crear un entorno desde cero. Asimismo, no está permitido utilizar los datos de producción en los entornos de desarrollo, prueba o aceptación.
 - C) Incorrecto. Los ficheros de configuración de los entornos deben ser distribuidos automáticamente.
 - D) Correcto. Esta es una de las opciones para automatizar la creación y la configuración del entorno. Los desarrolladores pueden utilizar la infraestructura programable para ellos mismos crear y configurar los entornos. (Bibliografía: A, Capítulo 9)

11 / 40

Un objetivo específico de diseño de un paquete de pruebas automatizadas es encontrar errores durante las pruebas lo más pronto posible. La pirámide ideal de automatización de pruebas muestra el orden correcto en el que deben realizarse las pruebas.

¿Qué prueba deberá realizarse **primero**?

- A) Prueba de APIs automatizada
- B) Prueba de componentes automatizada
- C) Prueba de integración automatizada
- D) Prueba unitaria automatizada

- A) Incorrecto. La prueba unitaria automatizada debe realizarse primero. Esta prueba es la cuarta en una secuencia de cinco.
- B) Incorrecto. La prueba unitaria automatizada debe realizarse primero. Esta prueba es la segunda en una secuencia de cinco.
- C) Incorrecto. La prueba unitaria automatizada debe realizarse primero. Esta prueba es la tercera a ejecutar en una secuencia de cinco.
- D) Correcto. Esta es la primera prueba que debe realizarse. (Bibliografía: A, Capítulo 10)

12 / 40

Un equipo de DevOps quiere aumentar la velocidad mediante la utilización de un desarrollo orientado a las pruebas.

¿Cuál es el orden correcto de las acciones?

- A) A) Refactorizar
B) Escribir un caso de prueba
C) Escribir el código funcional
- B) A) Escribir el código funcional
B) Escribir un caso de prueba
C) Refactorizar
- C) A) Escribir un caso de prueba
B) Escribir el código funcional
C) Refactorizar

- A) Incorrecto. En primer lugar, hay que escribir y ejecutar una prueba; después se tiene que implementar el código. La refactorización sería el tercer y último paso.
- B) Incorrecto. En primer lugar, hay que escribir y ejecutar una prueba; después se tiene que implementar el código; mientras que la refactorización es el tercer y último paso.
- C) Correcto. En primer lugar, hay que escribir y ejecutar una prueba; después se tiene que implementar el código. La refactorización es el tercer y último paso. (Bibliografía: A, Capítulo 10)

13 / 40

Una aseguradora contrata a un profesional de DevOps que asesore al equipo de DevOps para elegir una estrategia de ramificación. El profesional de DevOps evalúa dos estrategias:

1. Optimizar para la productividad individual
2. Optimizar para la productividad del equipo

¿Qué afirmación sobre estas dos estrategias es **verdadera**?

- A) Ambas estrategias dan lugar a la misma cantidad de esfuerzo para fusionar.
 - B) La estrategia 1 da lugar a mucho más esfuerzo para fusionar el código que la 2.
 - C) La estrategia 2 da lugar a mucho más esfuerzo para fusionar el código que la 1.
-
- A) Incorrecto. Puesto que la estrategia 1 da lugar a una rama por persona, la fusión del código se convierte en una pesadilla. Independientemente de la versión del sistema de gestión que se utiliza, la estrategia 2 es mucho mejor.
 - B) Correcto. La productividad individual supone una rama por desarrollador de DevOps, mediante la cual se puede producir software individualmente sin integrarlo. La estrategia 2 consiste en no tener rama alguna, por lo que el desarrollo se produce en una «línea continua». (Bibliografía: A, Capítulo 11)
 - C) Incorrecto. La estrategia 2 da lugar a menos esfuerzos para fusionar el código porque todo el mundo trabaja en una misma plataforma.

14 / 40

Una empresa minorista ha cambiado radicalmente el proceso de desarrollo desde un enfoque en cascada a un enfoque DevOps. Se deben tomar muchas decisiones en poco tiempo. Esto está causando una cierta deuda técnica. Por ejemplo, el tiempo de comercialización ha aumentado drásticamente. No obstante, existen maneras de eliminar muchos de los retrasos.

El mapeo de la cadena de valor muestra que el 20 % del tiempo del sprint se dedica a portar código, todo mantenido en ramas de código independientes.

¿Qué solución para este tipo de deuda técnica dará lugar a un flujo más rápido?

- A) Adoptar un desarrollo basado en rama principal
 - B) Reproducir las pruebas falladas en las estaciones de trabajo del desarrollador
 - C) Empezar a hacer *canary releases* (técnica de lanzamiento de bajo riesgo)
 - D) Usar más telemetría
-
- A) Correcto. El desarrollo basado en rama principal significa que la ramificación no está permitida y, por lo tanto, no hace falta fusionar el código. (Bibliografía: A, Capítulo 11)
 - B) Incorrecto. El problema es la ramificación, que causa problemas cuando se fusiona el código. Las pruebas no eliminan los retrasos.
 - C) Incorrecto. Las pruebas no eliminan los retrasos, ni siquiera en caso de lanzamientos de bajo riesgo, a pesar de su efecto positivo de cara al despliegue continuo.
 - D) Incorrecto. La telemetría no resuelve el problema de la fusión del código.

15 / 40

¿Cuál es una característica de una arquitectura que permite productividad, testabilidad y seguridad?

- A) Tener interfaces débilmente definidas
- B) Ser fuertemente acoplada
- C) Tener APIs bien definidas

- A) Incorrecto. Las interfaces bien definidas son una característica de una arquitectura que permite productividad, testabilidad y seguridad, al detallar de qué manera los módulos se conectan entre sí. Las interfaces débilmente definidas no lo son.
- B) Incorrecto. Una arquitectura fuertemente acoplada es monolítica o tiene módulos que están demasiado interconectados. Cada vez que intentamos confirmar un código en una rama principal, nos arriesgamos a crear fallos globales; cada pequeño cambio requiere cantidades enormes de comunicación y coordinación durante días, así como aprobaciones de cualquier grupo que podría verse potencialmente afectado.
- C) Correcto. Las interfaces de programación de aplicaciones (API) bien definidas permiten una arquitectura capaz de proporcionar productividad, pruebas más sencillas, y seguridad. (Bibliografía: A, Capítulo 13)

16 / 40

¿Cuál es la **mejor** estrategia de telemetría para resolver problemas cuando se trabaja con DevOps?

- A) Invertir en herramientas de telemetría es importante, y debería centrarse en el entorno de producción, *pipeline* de despliegue y preproducción.
 - B) Invertir en herramientas de telemetría es importante, y debería centrarse solo en el entorno de producción.
 - C) Invertir en herramientas de telemetría no es importante; la atención debería centrarse en reiniciar los servicios redundantes, lo cual es mucho más barato.
 - D) Invertir en herramientas de telemetría no es importante; la atención debería centrarse en el usuario de la aplicación, que es una fuente de información de monitorización mucho más completa y barata.
-
- A) Correcto. Los problemas no sólo ocurren en producción, sino también en otros entornos. DevOps requiere una retroalimentación rápida. La retroalimentación empieza en el inicio del *pipeline*. (Bibliografía: A, Capítulo 14)
 - B) Incorrecto. Centrarse únicamente en el entorno de producción no es suficiente. El *pipeline* de despliegue y el entorno de preproducción también son importantes para detectar errores pronto.
 - C) Incorrecto. Reiniciar los servicios no es más barato. Mucho tiempo de producción se va a perder intentando encontrar la causa del error.
 - D) Incorrecto. Si el usuario es la única fuente de monitorización del servicio, entonces la pérdida de tiempo de producción es elevada, ya que una monitorización automática bien implementada es mucho más rápida en localizar lo que está funcionando mal; además, no todos los errores pueden ser detectados por el usuario: sólo se podrán encontrar los errores que el usuario puede ver, y no los que están ocurriendo en segundo plano.

17 / 40

En la infraestructura de monitorización, los datos se deberían recopilar a partir de tres capas.

¿Cuál **no** es una de esas tres capas?

- A) Aplicación
- B) Lógica de negocio
- C) Métricas de negocio
- D) Sistema operativo

- A) Incorrecto. La capa de aplicación es una de las tres capas donde se debe realizar la recopilación de datos en la infraestructura de monitorización.
- B) Incorrecto. La capa de lógica de negocio es una de las tres capas donde se debe realizar la recopilación de datos en la infraestructura de monitorización.
- C) Correcto. Las métricas de negocio no son una de las tres capas donde se debe realizar la recopilación de datos en la infraestructura de monitorización. Esos indicadores son más bien el resultado de la monitorización. (Bibliografía: A, Capítulo 14)
- D) Incorrecto. La capa de sistema operativo es una de las tres capas donde se debe realizar la recopilación de datos en la infraestructura de monitorización.

18 / 40

¿Para qué tipo de destinatarios aporta valor el acceso de autoservicio a la telemetría?

- A) Sólo desarrolladores
- B) Sólo desarrolladores y operadores
- C) Sólo desarrolladores, operadores y partes interesadas
- D) Desarrolladores, operadores, partes interesadas y clientes

- A) Incorrecto. También aporta valor para los clientes, las partes interesadas y los operadores.
- B) Incorrecto. También aporta valor para los clientes y las partes interesadas.
- C) Incorrecto. También aporta valor para los clientes.
- D) Correcto. El acceso de autoservicio a la telemetría aporta valor a todos los visitantes. (Bibliografía: A, Capítulo 14)

19 / 40

En un entorno que tiene pruebas automatizadas, procesos de despliegue rápido y telemetría adecuada, ¿qué técnica ofrece **mayores** beneficios de DevOps al negocio?

- A) Hacer una corrección preventiva (*fix forward*)
 - B) Restaurar la última versión (*roll back*)
 - C) Retirar los servidores averiados
 - D) Desactivar las funcionalidades que fallan
-
- A) Correcto. Aunque esto podría ser una elección arriesgada, es totalmente segura en este entorno que tiene pruebas automatizadas, procesos de despliegue rápido y telemetría adecuada. Gracias a eso, se podrá confirmar rápidamente si todo funciona de manera correcta en producción y se podrán ofrecer nuevas funcionalidades y valor al negocio. El valor añadido consiste en que los errores se corrigen en vez de eliminar la nueva funcionalidad o retirar los servidores averiados. (Bibliografía: A, Capítulo 16)
 - B) Incorrecto. Al igual que «desactivar las funcionalidades que fallan», restaurar la última versión no hace más que devolver el despliegue al estado anterior y eliminar los problemas desactivando lo que provocó el error. Por lo tanto, no se repara el error y, además, se quita valor de negocio del entorno de producción.
 - C) Incorrecto. Esta opción se centra en la continuidad del servicio, pero dado que podemos contar con procesos, telemetría y pruebas adecuados, la mejor opción es ofrecer nuevo valor de negocio.
 - D) Incorrecto. Desactivar las funcionalidades que fallan es la opción menos arriesgada, pero no corrige el error y quita valor de negocio del entorno de producción.

20 / 40

Una empresa de software ha redactado los siguientes requisitos para la guía de lanzamientos:

- Recuento y gravedad de defectos: ¿La aplicación funciona conforme al diseño?
- Cobertura de monitorización: ¿La cobertura de monitorización es suficiente para restaurar el servicio cuando las cosas no vayan bien?
- Arquitectura de sistemas: ¿Está el servicio lo suficientemente acoplado como para resolver una tasa elevada de incidentes en producción?
- Tipo/frecuencia de mensajes de alerta a buscaperonas: ¿La aplicación está generando un número insostenible de alertas en producción?

¿Qué requisito de la guía de lanzamientos **no** cumple con la forma de trabajar de DevOps?

- A) Recuento y gravedad de defectos: ¿La aplicación funciona realmente conforme al diseño?
- B) Cobertura de monitorización: ¿La cobertura de monitorización es suficiente para restaurar el servicio cuando las cosas no vayan bien?
- C) Arquitectura de sistemas: ¿Está el servicio lo suficientemente acoplado como para resolver una tasa elevada de incidentes en producción?
- D) Tipo/frecuencia de mensajes de alerta a buscaperonas: ¿La aplicación está generando un número insostenible de alertas en producción?

- A) Incorrecto. Este requisito sí que cumple con la forma de trabajar de DevOps.
- B) Incorrecto. Este requisito sí que cumple con la forma de trabajar de DevOps.
- C) Correcto. Cuando se trabaja con DevOps, la arquitectura debería estar débilmente acoplada y no fuertemente acoplada. El servicio debería estar débilmente acoplado para poder soportar un índice elevado de cambios y despliegues en producción. (Bibliografía: A, Capítulo 16)
- D) Incorrecto. Este requisito sí que cumple con la forma de trabajar de DevOps.

21 / 40

¿Cuál es una característica única de la revisión de preparación para el traspaso (HRR) y **no** de la revisión de preparación para el lanzamiento (LRR)?

- A) La HRR es mucho más estricta y tiene estándares de aceptación más elevados.
- B) La HRR es autoevaluada por los equipos de producto.
- C) La HRR se aprueba antes de que cualquier nuevo servicio se ponga a disposición del público.
- D) La HRR se aprueba antes de que cualquier nuevo servicio reciba tráfico de producción activo.

- A) Correcto. Ésta es una característica única de HRR, ya que la revisión se aprueba cuando el servicio se transfiere a un estado gestionado por Operaciones. (Bibliografía: A, Capítulo 16)
- B) Incorrecto. Ésta es una característica única de LRR, no de HRR.
- C) Incorrecto. Ésta es una característica única de LRR, no de HRR.
- D) Incorrecto. Ésta es una característica única de LRR, no de HRR.

22 / 40

Una de las técnicas más potentes en el diseño de interacción y experiencia de usuario (UX) es la investigación contextual.

¿Cuál es la **mejor** descripción de investigación contextual?

- A) El equipo de producto pide a los usuarios que respondan a una demostración de la aplicación por parte del equipo de producto.
 - B) El equipo de producto entrevista a los usuarios sobre el uso de la aplicación en sus hogares.
 - C) El equipo de producto observa a los usuarios que utilizan la aplicación en su entorno natural.
 - D) El equipo de producto estudia a los usuarios durante la prueba de aceptación de usuario en una sala especial con equipamientos de prueba.
-
- A) Incorrecto. La investigación contextual consiste en observar cómo el usuario utiliza la aplicación, y no en una demostración de la aplicación por parte del proveedor.
 - B) Incorrecto. La investigación contextual consiste en observar cómo el usuario utiliza la aplicación, y no en una entrevista sobre el uso de la aplicación.
 - C) Correcto. Esta técnica consiste en observar cómo el cliente utiliza la aplicación en su entorno habitual. (Bibliografía: A, Capítulo 16)
 - D) Incorrecto. La investigación contextual consiste en observar cómo el usuario utiliza la aplicación en su entorno habitual, y no en una sala especial con equipamientos de prueba.

23 / 40

La startup ABC está teniendo serios problemas en el desarrollo de las funcionalidades para una aplicación de navegación móvil que satisfaga las expectativas del usuario.

¿Cuál es la **mejor** estrategia para garantizar que las funcionalidades estén directamente alineadas con las expectativas del usuario respecto a la experiencia de usuario de la aplicación de navegación móvil?

- A) Desarrollar más plenamente la experiencia de usuario de la aplicación de navegación móvil, de manera que los consumidores dispongan de funcionalidades y opciones más avanzadas para navegar en la experiencia de usuario, ofreciéndoles más opciones para personalizar la misma.
 - B) Desarrollar la aplicación de navegación móvil basada en la experiencia de usuario, en la que se seleccionan aleatoriamente a consumidores para mostrarles una de dos versiones de una experiencia de usuario entre las que pueden elegir, o bien la versión de referencia (control) o una variante.
 - C) Efectuar correcciones sobre las características actuales relacionadas con la experiencia de usuario de la aplicación de navegación móvil, de manera que ABC se ciña a la funcionalidad básica de la experiencia de usuario sin sacrificar la calidad.
 - D) Lanzar al mercado una funcionalidad de la aplicación de navegación móvil para recibir la opinión de los consumidores sobre la funcionalidad actual durante un período de tres semanas.
-
- A) Incorrecto. Cuando descubrimos que una funcionalidad determinada no está logrando los resultados deseados, añadir más funcionalidades o mejoras a la misma podría tener menos prioridad que implementar otras funcionalidades, lo cual implica que esa funcionalidad que no está teniendo el comportamiento esperado podría no alcanzar nunca su objetivo de negocio.
 - B) Correcto. La técnica A/B más utilizada en la práctica actual de la experiencia de usuario consiste en un sitio web donde los visitantes son seleccionados aleatoriamente para mostrarles una de las dos versiones de una página, o bien la versión de control (la «A») o una variante (la «B»). Basándonos en el análisis estadístico del comportamiento posterior de estos dos grupos de usuarios, demostramos si existe una diferencia significativa en los resultados de las dos, estableciendo un vínculo causal entre la variante y los resultados. (Bibliografía: A, capítulo 17).
 - C) Incorrecto. Cuando descubrimos que una funcionalidad determinada no está logrando los resultados deseados, efectuar correcciones en dicha funcionalidad podría tener menos prioridad que implementar nuevas funcionalidades, lo cual implica que esa funcionalidad que no está teniendo el comportamiento esperado podría no alcanzar nunca su objetivo de negocio.
 - D) Incorrecto. Cada experimento normalmente requiere mucho tiempo de ensayo, así que, para completarlo, se tardan semanas o incluso meses.

24 / 40

¿Qué estrategia puede utilizar un equipo de desarrollo para ayudar a conseguir los resultados de negocio esperados?

- A) Ejecución con esmero numerosos experimentos
- B) Pruebas de regresión completas
- C) Desarrollo guiado por hipótesis
- D) SDLC (ciclo de vida de desarrollo de software)

- A) Incorrecto. Ésta no es una estrategia para intentar alcanzar los resultados de negocio esperados.
- B) Incorrecto. Ésta no es una estrategia para intentar alcanzar los resultados de negocio esperados.
- C) Correcto. Ésta es la mejor estrategia para intentar alcanzar los resultados de negocio esperados. (Bibliografía: A, capítulo 17).
- D) Incorrecto. Ésta no es una estrategia para intentar alcanzar los resultados de negocio esperados.

25 / 40

La empresa ABC ha tenido dificultades para comprender el impacto de los cambios en el código en su aplicación de nóminas. Actualmente, los cambios implementados en el código están mal documentados. Por ejemplo, la última corrección de código se documentó como se indica a continuación: «Corrección problema n.º 1801 para la aplicación de nóminas». La empresa se da cuenta de que ésta es una *pull request* mala.

Una buena *pull request* incluye lo siguiente:

- ¿Por qué implementamos la corrección?
- ¿Quién implementa la corrección?

¿Qué más se **debe** incluir en una buena *pull request*?

- A) Las unidades de negocio afectadas por la corrección
- B) Los posibles riesgos y contramedidas relacionados con la implementación de la corrección
- C) El plan de apoyo al lanzamiento para implementar la corrección

- A) Incorrecto. Se deben incluir los posibles riesgos y contramedidas relacionados con la implementación de la corrección.
- B) Correcto. Una *pull request* efectiva debe proporcionar detalles sobre el motivo por el que implementamos la corrección, quién la implementa y los posibles riesgos y contramedidas relacionados con la implementación de la corrección. (Bibliografía: A, Capítulo 18)
- C) Incorrecto. Se deben incluir los posibles riesgos y contramedidas relacionados con la implementación de la corrección.

26 / 40

Considere el siguiente escenario:

«El desarrollador revisa el código con un compañero que le da su opinión al respecto.».

¿Qué técnica de revisión se describe aquí?

- A) Revisión supervisada por un par (*over-the-shoulder*)
- B) Programación en parejas
- C) Revisión por pares
- D) Revisión asistida por herramientas

- A) Correcto. La técnica de revisión supervisada por un par consiste en la interacción entre el autor y un compañero después de escribir el código. (Bibliografía: A, Capítulo 18)
- B) Incorrecto. Esta técnica de revisión se realiza durante la codificación y no después.
- C) Incorrecto. Esto se realiza después, pero no es necesario que el autor esté presente.
- D) Incorrecto. Esta revisión no la realiza una persona, sino una herramienta.

27 / 40

¿Qué técnica de revisión resulta directamente en una disminución de los errores de codificación?

- A) Revisión supervisada por un par (*over-the-shoulder*)
- B) Programación en parejas
- C) Revisión por pares
- D) Revisión asistida por herramientas

- A) Incorrecto. Esto permite recibir retroalimentación mientras se explora el software que ya está escrito.
- B) Correcto. La programación en parejas significa escribir software junto con un compañero. Una persona escribe y la otra revisa. Esto resulta en retroalimentación directa y menos errores. (Bibliografía: A, Capítulo 18)
- C) Incorrecto. La revisión por pares es una revisión que se hace después de escribir el software.
- D) Incorrecto. La revisión asistida por herramientas es limitada en cuanto a retroalimentación y muchas comprobaciones se hacen después de subir el código.

28 / 40

Un proveedor de servicios en la nube quiere aumentar la capacidad de su servicio utilizando un mono del ejército de los simios (*The Simian Army*).

¿Qué mono es necesario en este caso?

- A) Mono Doctor (*Doctor Monkey*)
- B) Mono Conserje (*Janitor Monkey*)
- C) Mono Latencia (*Latency Monkey*)

- A) Incorrecto. El Mono Doctor accede a los controles de salud que se ejecutan por cada instancia y, cuando encuentra algo que está mal, cierra proactivamente las instancias correspondientes si los responsables no han corregido la causa raíz a tiempo. Esto no mejora la capacidad.
- B) Correcto. El Mono Conserje se asegura de que su entorno en la nube está funcionando sin desorden ni ineficiencias; busca recursos no utilizados y los elimina. (Bibliografía: A, Apéndice 9)
- C) Incorrecto. El Mono Latencia induce retrasos artificiales o inactividad en su capa de comunicaciones RESTful entre el cliente y el servidor para simular degradaciones de servicio y garantizar que los servicios dependientes responden adecuadamente. Esto no mejora la capacidad.

29 / 40

La empresa ABC está adoptando DevOps en su manera de trabajar y quiere fomentar un entorno de aprendizaje que sea abierto y donde no se culpabilice a nadie. ABC sufrió recientemente un fallo importante en una aplicación y pudo restaurar el servicio.

¿Cuál es la **primera** tarea que debe completarse durante la reunión de análisis *a posteriori* libre de culpa?

- A) Crear un cronograma de eventos relevantes tal y como ocurrieron durante el fallo de la aplicación
- B) Identificar contramedidas para evitar que el fallo de la aplicación vuelva a ocurrir en el futuro
- C) Identificar la causa raíz del fallo de la aplicación para proponer una medida correctiva con el fin de evitar que el mismo fallo vuelva a ocurrir en el futuro
- D) Publicar los resultados del análisis *a posteriori* en una ubicación centralizada de manera que toda la organización pueda acceder a ellos y aprender del fallo de la aplicación

- A) Correcto. La primera tarea en la reunión de análisis *a posteriori* libre de culpa es registrar el mejor entendimiento posible del cronograma de los eventos relevantes tal y como ocurrieron. Esto incluye todas las medidas adoptadas y el momento en que se adoptaron (idealmente, respaldadas por registros de chat, como IRC o Slack), qué repercusiones se han observado (lo ideal sería en forma de las métricas específicas de nuestra telemetría de producción, en lugar de descripciones meramente subjetivas), todas las rutas de investigación que se han seguido y las resoluciones que han tomado en consideración. (Bibliografía: A, Capítulo 19)
- B) Incorrecto. La identificación de contramedidas no es la primera tarea para llevar a cabo en una reunión de análisis *a posteriori* libre de culpa. Ésta es una acción que se podría producir después de crear el cronograma.
- C) Incorrecto. Determinar la causa raíz no es la primera tarea para llevar a cabo en una reunión de análisis *a posteriori* libre de culpa. Ésta es una acción que se podría producir después de crear el cronograma.
- D) Incorrecto. Publicar los resultados del análisis *a posteriori* no es la primera tarea para llevar a cabo en una reunión de análisis *a posteriori* libre de culpa. Ésta es una acción que se podría producir sólo después de documentar por completo el análisis *a posteriori*.

30 / 40

¿Qué es necesario cuando se crea resiliencia inyectando fallos en producción?

- A) Definir el modo de fallo
- B) Organizar reuniones de análisis *a posteriori*
- C) Formar al personal
- D) Usar un entorno de prueba

- A) Correcto. La definición del modo de fallo es importante para garantizar que estos modos de fallo funcionan según el diseño. (Bibliografía: A, Capítulo 19)
- B) Incorrecto. Las reuniones de análisis *a posteriori* no son parte de las técnicas de inyección.
- C) Incorrecto. El personal aprende de la inyección.
- D) Incorrecto. Sólo es necesario un entorno de producción.

31 / 40

¿Cuál es el **primer** paso de un plan para utilizar gamificación para simular accidentes (*Game Days*)?

- A) Definir y ejecutar simulacros
 - B) Identificar y abordar problemas, y probar
 - C) Planificar la interrupción del servicio
 - D) Preparar y eliminar los puntos únicos de fallo
-
- A) Incorrecto. El primer aspecto es planificar: «Planificar la interrupción del servicio de una planta mediante una inyección de fallos a gran escala». «Definir el plan y ejecutar simulacros» es una tarea para los responsables del simulacro gamificado.
 - B) Incorrecto. El primer aspecto es planificar: «Planificar la interrupción del servicio de una planta mediante una inyección de fallos a gran escala».
 - C) Correcto. Los pasos deben ser los siguientes:
 1. El primer paso es la planificación de la interrupción del servicio
 2. Basándose en el plan, el equipo puede preparar la interrupción del servicio adoptando medidas
 3. La medida puede suponer procedimientos que es necesario probar
 4. La interrupción del servicio se ejecuta según lo previsto
 5. Se debe seguir el proceso definido. (Bibliografía: A, Capítulo 19)
 - D) Incorrecto. El primer aspecto es planificar: «Planificar la interrupción del servicio de una planta mediante una inyección de fallos a gran escala».

32 / 40

¿Cuál es un ejemplo de un requisito no funcional?

- A) Tener compatibilidad con versiones anteriores y posteriores
 - B) Permitir informar sobre retrasos en los planes
 - C) Registrar transacciones financieras para un sistema de reservas de hotel
-
- A) Correcto. La compatibilidad con versiones anteriores y posteriores es un ejemplo de un requisito no funcional. (Bibliografía: A, Capítulo 20)
 - B) Incorrecto. El informe es una función y, por lo tanto, no es un requisito no funcional.
 - C) Incorrecto. Las transacciones financieras son una funcionalidad. Los requisitos no funcionales están relacionados con la calidad del sistema.

33 / 40

¿Qué actividad se debería realizar para la creación satisfactoria de historias de usuario de operaciones reutilizables?

- A) Asociar las historias de usuario de operaciones a las mejoras y defectos de desarrollo pertinentes
 - B) Definir actividades dentro del proceso de traspaso y posteriormente automatizar dichas actividades mediante el uso de herramientas y flujos de trabajo de apoyo adecuados
 - C) Identificar todas las actividades de trabajo operativo necesarias y los actores necesarios para completarlas
-
- A) Incorrecto. Ésta no es una actividad que se debe tener en cuenta para la creación satisfactoria de historias de usuario de Operaciones reutilizables.
 - B) Correcto. Ésta es una actividad que se debe tener en cuenta para la creación satisfactoria de historias de usuario de Operaciones reutilizables. (Bibliografía: A, Capítulo 20)
 - C) Incorrecto. Ésta no es una actividad que se debe tener en cuenta para la creación satisfactoria de historias de usuario de Operaciones reutilizables.

34 / 40

Considere los siguientes elementos:

1. Solicitudes de cambio
2. Herramientas del *pipeline* de despliegue
3. Ejecutables de programas compilados
4. Tutoriales y estándares

¿Qué **dos** elementos se suelen guardar en un repositorio único de código fuente compartido?

- A) 1 y 2
- B) 1 y 4
- C) 2 y 3
- D) 2 y 4

- A) Incorrecto. Las solicitudes de cambio (1) no forman parte del repositorio único de código fuente compartido, ya que no son artefactos que codifican conocimientos y aprendizaje. Las solicitudes de cambio se registran en el backlog de producto, que no forma parte del repositorio de código fuente compartido. Las herramientas del *pipeline* de despliegue (2) forman parte del repositorio de código fuente compartido único.
- B) Incorrecto. Las solicitudes de cambio (1) no forman parte del repositorio único de código fuente compartido, ya que no son artefactos que codifican conocimientos y aprendizaje. Las solicitudes de cambio se registran en el backlog de producto, que no forma parte del repositorio de código fuente compartido. Los tutoriales y estándares (4) forman parte del repositorio único de código fuente compartido.
- C) Incorrecto. Los ejecutables de programas compilados (3) no forman parte del repositorio de código fuente compartido, ya que no son artefactos que codifican conocimientos y aprendizaje. Los ejecutables se guardan como archivos binarios en el almacén de artefactos (salida del *pipeline* de despliegue) y no en el repositorio único de código fuente compartido (entrada del *pipeline* de despliegue). Las herramientas del *pipeline* de despliegue (2) forman parte del repositorio único de código fuente compartido.
- D) Correcto. Tanto las herramientas del *pipeline* de despliegue (2) como los tutoriales y estándares (4) forman parte del repositorio único de código fuente compartido, ya que son artefactos que codifican conocimientos y aprendizaje. (Bibliografía: A, Capítulo 20)

35 / 40

¿Cuál es el objetivo de convertir descubrimientos locales en mejoras globales?

- A) Elevar el nivel de las prácticas vigentes, no solo de Desarrollo y Operaciones, sino de toda la organización
- B) Hacer que el aprovechamiento del conocimiento colectivo sea más sencillo, tanto para los servicios nuevos como para los ya existentes
- C) Hacer que la cultura de trabajo sea más colaborativa, y que los sistemas sean más seguros y resilientes
- D) Reforzar una cultura en la que todos se sientan cómodos y responsables

- A) Correcto. Éste es el objetivo de convertir descubrimientos locales en mejoras globales. (Bibliografía: A, Capítulo 20)
- B) Incorrecto. Éste es un objetivo del diseño subyacente de Operaciones a través de requisitos no funcionales codificados.
- C) Incorrecto. Éste es un objetivo de reservar tiempo para crear mejoras y aprendizaje organizativos.
- D) Incorrecto. Éste es un objetivo de hacer posible e introducir aprendizaje en el trabajo diario.

36 / 40

Los desarrolladores pueden facilitar que cualquier ingeniero cree y utilice de forma correcta las normas de registro y de criptografía en sus aplicaciones y entornos.

¿Qué opción **no** es un elemento del repositorio de código fuente compartido compatible con esto?

- A) Librerías de código y sus configuraciones recomendadas
- B) Paquetes de despliegue
- C) Paquetes y compilaciones de sistemas operativos (SO)
- D) Herramientas de gestión de preguntas de seguridad

- A) Incorrecto. Dado que las librerías de código y sus configuraciones recomendadas tienen ajustes de configuración eficaces en materia de seguridad para los componentes que utilizan en su aplicación, eso facilita que cualquier ingeniero pueda crear y utilizar de forma correcta las normas de acceso y de criptografía.
- B) Correcto. Los paquetes de despliegue son un producto entregable de un ingeniero (Operaciones) y no de un desarrollador (Desarrollo); por lo tanto, no soportan esto. La pregunta se refiere a lo que un desarrollador puede hacer para facilitar que cualquier ingeniero pueda crear y utilizar de forma correcta las normas de acceso y de criptografía en sus aplicaciones. (Bibliografía: A, Capítulo 22)
- C) Incorrecto. Los paquetes y compilaciones de sistemas operativos facilitan que cualquier ingeniero pueda crear y utilizar de forma correcta las normas de acceso y de criptografía en sus aplicaciones, ya que proporcionan ajustes de configuración eficaces en materia de seguridad para los componentes que utilizan en su aplicación.
- D) Incorrecto. Las herramientas de gestión de preguntas de seguridad facilitan que cualquier ingeniero pueda crear y utilizar de forma correcta las normas de acceso y de criptografía en sus aplicaciones, ya que proporcionan ajustes de configuración eficaces en materia de seguridad, tal como ajustes de conexión, claves de cifrado, etcétera.

37 / 40

Cuando los desarrolladores introducen código, siempre existe el riesgo de permitir un acceso no autorizado.

¿Qué control **no** mitiga ese riesgo?

- A) Revisiones de código
- B) Pruebas de código
- C) Parches efectivos
- D) Pruebas de intrusión

- A) Incorrecto. La revisión de código puede revelar la introducción de código malicioso que podría permitir el acceso al software a través de una puerta trasera.
- B) Incorrecto. Las pruebas de código pueden revelar la habilitación de un acceso no autorizado.
- C) Correcto. Los parches efectivos no revelan la introducción de código del desarrollador; solo resuelven errores. (Bibliografía: A, Capítulo 22)
- D) Incorrecto. Las pruebas de intrusión pueden revelar puntos débiles en la aplicación que se hayan creado o dejado allí a propósito para permitir accesos no autorizados.

38 / 40

¿Cuál es un ejemplo de creación de telemetría para una aplicación?

- A) Cambios en el sistema operativo (SO)
- B) Revisión diaria de los registros de sistema
- C) Cambios en los grupos de seguridad
- D) Restablecimientos de las contraseñas de los usuarios

- A) Incorrecto. Éste es un uso de la telemetría para el entorno de la infraestructura donde está alojada la aplicación, y no específicamente para la aplicación.
- B) Incorrecto. Éste es un uso de la telemetría para el entorno de la infraestructura donde está alojada la aplicación, y no específicamente para la aplicación.
- C) Incorrecto. Éste es un uso de la telemetría para el entorno de la infraestructura donde está alojada la aplicación, y no específicamente para la aplicación.
- D) Correcto. Éste es un uso específico de la telemetría para una aplicación. (Bibliografía: A, Capítulo 22)

39 / 40

¿Qué proceso sirve como control primario para reducir los riesgos de seguridad y de Operaciones, y también soporta los requisitos de conformidad?

- A) Proceso de gestión de cambios
- B) Proceso de gestión de la configuración
- C) Proceso de gestión de entregas y despliegues
- D) Proceso de gestión del nivel de servicio

- A) Correcto. Casi cualquier organización de TI de un tamaño relevante dispondrá de procesos vigentes de gestión de cambios, que son los controles primarios para reducir los riesgos de seguridad y de operaciones. Los directores de conformidad y de seguridad depositan su confianza en los procesos de gestión de cambios para los requisitos de conformidad y normalmente exigen evidencias de que todos los cambios han sido debidamente autorizados. (Bibliografía: A, Capítulo 23)
- B) Incorrecto. Este proceso proporciona información sobre los servicios y cada elemento de configuración. Esto podría utilizarse para analizar y detectar posibles riesgos y mejoras, pero no se trata del proceso responsable de reducir esos riesgos. La gestión de la configuración planifica, implementa, controla, notifica y verifica los elementos de configuración.
- C) Incorrecto. La gestión de versiones y despliegues es el proceso que realmente implementa la versión aprobada que a su vez puede incluir numerosos cambios aprobados. El proceso de gestión de cambios controla y aprueba los cambios y después coordina esfuerzos con el proceso de gestión de versiones y despliegues.
- D) Incorrecto. El proceso de gestión del nivel de servicio garantiza la entrega del nivel de servicio requerido, según lo acordado con el negocio. Establece qué requisitos de seguridad y conformidad son necesarios, pero no es el proceso que contribuirá a reducir esos riesgos.

40 / 40

¿Qué supone un inconveniente a la hora de adoptar la segregación de funciones como medida de control?

- A) La segregación de funciones como medida de control puede dificultar a menudo los esfuerzos de desarrollo al ralentizar y reducir la retroalimentación que reciben los ingenieros sobre su trabajo.
 - B) La segregación de funciones como medida de control requiere que el desarrollador envíe los cambios al responsable de la librería de código, quien revisará y aprobará el cambio antes de que éste pase a producción.
 - C) La segregación de funciones como medida de control da lugar a trabajo innecesario, ya que requiere la inspección del código que se sube, y las revisiones de código, que proporcionan la confianza necesaria sobre la calidad de nuestro trabajo.
-
- A) Correcto. La separación de funciones a menudo puede dificultar esto al ralentizar y reducir la retroalimentación que reciben los ingenieros sobre su trabajo. Esto impide que los ingenieros asuman la plena responsabilidad de la calidad de su trabajo y reduce la capacidad de una empresa para crear aprendizaje organizativo. Por consiguiente, siempre que sea posible, deberíamos evitar el uso de la separación de funciones como control. En lugar de eso, deberíamos elegir controles como la programación en parejas, la inspección continua de las subidas de código, y la revisión de código. (Bibliografía: A, Capítulo 23)
 - B) Incorrecto. Esto refleja el antiguo enfoque que apoyaba al ciclo de vida de desarrollo de software y no a DevOps. Los controles recomendados para trabajar en ámbito DevOps incluyen la programación en parejas, la inspección continua de las subidas de código, y la revisión de código. Estos controles nos proporcionan la confianza necesaria sobre la calidad de nuestro trabajo. Asimismo, al aplicar estos controles, si es necesaria la segregación de funciones, podemos demostrar que logramos resultados equivalentes con los controles que hemos creado.
 - C) Incorrecto. Deberíamos evitar el uso de la segregación de funciones como medida de control. En lugar de eso, deberíamos elegir controles como la programación en parejas, la inspección continua de las subidas de código, y la revisión de código. Asimismo, al aplicar estos controles, si es necesaria la segregación de funciones, podemos demostrar que logramos resultados equivalentes con los controles que hemos creado.

Evaluación

En la siguiente tabla se indican las respuestas correctas a las preguntas.

Número	Respuesta	Número	Respuesta
1	C	21	A
2	A	22	C
3	B	23	B
4	A	24	C
5	B	25	B
6	A	26	A
7	B	27	B
8	C	28	B
9	C	29	A
10	D	30	A
11	D	31	C
12	C	32	A
13	B	33	B
14	A	34	D
15	C	35	A
16	A	36	B
17	C	37	C
18	D	38	D
19	A	39	A
20	C	40	A



Contacto EXIN

www.exin.com

