
RESPUESTAS EXAMEN MODELO TIPO C

1. Pregunta 1 (1 Punto)

¿Cuál de los siguientes es un objetivo de prueba típico?

- a) Validar que se cumplan los requisitos documentados.
- b) Provocar averías e identificar defectos.
- c) Iniciar errores e identificar las causas fundamentales.
- d) Verificar que el objeto de prueba cumpla con las expectativas del usuario.

Seleccione una opción.

Solución

- a) *No es correcto. Validar que se cumplan los requisitos documentados es incorrecto ya que la validación tiene que ver con cumplir con los requisitos y expectativas del usuario, mientras que la verificación tiene que ver con cumplir con requisitos específicos, por lo que esto sería correcto si reemplazáramos “validar” por “verificar”.*
- b) *Es correcto. Provocar fallos e identificar defectos es probablemente el objetivo más común de las pruebas dinámicas.*
- c) *No es correcto. Iniciar errores e identificar las causas fundamentales es incorrecto porque los evaluadores no inician errores, sino que intentan provocar fallas. Los errores generalmente los cometen los desarrolladores (y en realidad no pueden iniciarse) y resultan en defectos, que los evaluadores intentan identificar directamente mediante pruebas estáticas o indirectamente mediante fallas en las pruebas dinámicas. Identificar las causas fundamentales es útil, pero forma parte de la depuración, que es una actividad independiente de las pruebas.*
- d) *No es correcto. Verificar que el objeto de prueba cumpla con las expectativas del usuario es incorrecto ya que la verificación se refiere a verificar que se cumplan los requisitos especificados (documentados), mientras que la validación se refiere a cumplir con los requisitos y expectativas del usuario, por lo que esto sería correcto si reemplazáramos “verificar” por “validar”.*

2. Pregunta 2 (1 Punto)

¿Cuál de las siguientes afirmaciones describe mejor la diferencia entre probar y depurar?

- a) Las pruebas causan fallas, mientras que la depuración corrige las fallas.
- b) Las pruebas son una actividad negativa, mientras que la depuración es una actividad positiva.
- c) Las pruebas determinan que existen defectos, mientras que la depuración elimina los defectos.
- d) Las pruebas encuentran la causa de los defectos mientras que la depuración soluciona la causa de los defectos.

Seleccione una opción.

Solución

- a) *No es correcto. Las pruebas dinámicas causan fallas (a partir de las cuales se pueden localizar y reparar los defectos). Sin embargo, la depuración se ocupa de localizar defectos y corregirlos. Por lo tanto, la depuración no soluciona los fallos.*
- b) *No es correcto. Tanto las pruebas como la depuración contribuyen a mejorar la calidad del objeto de prueba, por lo que ambas deberían considerarse positivamente. La depuración generalmente se considera una actividad positiva, ya que consiste en arreglar algo. Las pruebas dinámicas implican hacer que el objeto de prueba falle intencionalmente, razón por la cual algunas personas lo consideran una actividad negativa, pero esa es una visión muy limitada (y no es la que suelen tener los evaluadores). Son posibles casos de prueba tanto positivos como negativos. Los casos de prueba positivos verifican que el objeto de prueba realiza correctamente lo que se supone que debe hacer, mientras que las pruebas negativas verifican que el objeto de prueba no hace lo que se supone que no debe hacer.*
- c) *Es correcto. Las pruebas determinan que existen defectos, ya sea directamente a través de la observación del defecto en las revisiones (o mediante una herramienta de análisis estático), o indirectamente al causar una falla en las pruebas dinámicas. La depuración es una actividad separada de las pruebas (normalmente realizadas por desarrolladores) y se ocupa de localizar defectos (solo para pruebas dinámicas) y corregirlos.*
- d) *No es correcto. Las causas de los defectos suelen ser errores humanos. Las pruebas encuentran defectos directamente mediante pruebas estáticas o indirectamente provocando fallas en las pruebas dinámicas, y la depuración corrige los defectos. Entonces, las pruebas no encuentran la causa de los defectos y la depuración no soluciona las causas de los defectos.*

Mirar sección 2.1.2.

3. Pregunta 3 (1 Punto)

La “falacia de ausencia de defectos” es uno de los principios de las pruebas. ¿Cuál de los siguientes es un ejemplo de cómo abordar este principio en la práctica?

- a) Hay que explicar que no es posible que las pruebas demuestren la ausencia de defectos.
- b) Apoyar a los usuarios finales para realizar pruebas de aceptación.
- c) Hay que asegurar que no queden defectos de implementación en el sistema entregado.
- d) Modificar las pruebas que no causan fallas para garantizar que queden pocos defectos.

Solución

La “falacia de ausencia de defectos” tiene que ver con la idea de que garantizar la corrección de acuerdo con los requisitos (es decir, verificar la ausencia de defectos de implementación) no garantiza la satisfacción del usuario con el sistema. Para abordar esto, también es necesario validar que el sistema cumpla con las necesidades y expectativas de los usuarios, cumpla con los objetivos comerciales y supere a los sistemas de la competencia.

- a) No es correcto. El principio de “las pruebas muestran la presencia, no la ausencia de defectos” explica que si bien las pruebas pueden detectar la existencia de defectos en el objeto de prueba, no es posible demostrar que no los hay y, por lo tanto, garantizar su corrección. Por lo tanto, hay que explicar que no es posible que las pruebas demuestren la ausencia de defectos abordaría parcialmente este principio, no la falacia de la “ausencia de defectos”.*
- b) Es correcto. Al ayudar al usuario final a realizar pruebas de aceptación, debería ser posible validar que el sistema satisfaga las necesidades y expectativas de los usuarios.*
- c) No es correcto. No es posible garantizar que no queden defectos de implementación en el sistema entregado, ya que el principio de “las pruebas muestran la presencia, no la ausencia de defectos” explica que, si bien las pruebas pueden detectar la existencia de defectos en el objeto de prueba, no es posible demostrar que no existen defectos y, por tanto, garantizar su corrección.*
- d) No es correcto. Modificar las pruebas que no causan fallas para garantizar que queden pocos defectos es una forma de abordar el principio de “desgaste de las pruebas”. Este principio tiene que ver con la idea de que es poco probable que la repetición de pruebas idénticas en código inalterado descubra defectos nuevos y, por lo tanto, modificar las pruebas puede ser esencial. Esto no validará que el sistema cumpla con las necesidades y expectativas de los usuarios.*

Mirar sección 1.3.1.

4. Pregunta 4 (1 Punto)

¿Cuál de las siguientes actividades de prueba es más probable que implique la aplicación del análisis de valores límite y la partición de equivalencia?

- a) Implementación de pruebas.
- b) Diseño de prueba.
- c) Ejecución de la prueba.
- d) Monitoreo de pruebas.
- e) Análisis de pruebas.

Seleccione dos opciones.

Solución

Dada la siguiente descripción del análisis de prueba: para identificar las características que requieren prueba, la base de la prueba se analiza y define como condiciones de prueba, que luego se priorizan junto con los riesgos relacionados. La identificación sistemática de condiciones de prueba como elementos de cobertura a menudo implica el uso de técnicas de prueba tanto durante el análisis de la prueba como parte de la actividad de diseño de la prueba. De la descripción anterior, se puede ver que las técnicas de prueba se utilizan a menudo en las actividades de análisis y diseño de pruebas. El análisis de valores límite y la partición de equivalencia son técnicas de prueba.

- a) *No es correcto. No es probable que la implementación de pruebas implique el uso de técnicas de prueba, ya que se ocupa principalmente de ensamblar casos de prueba en procedimientos de prueba, mientras que las técnicas de prueba crean casos de prueba.*
- b) *Es correcto. Es probable que el diseño de pruebas implique el uso de técnicas de prueba para crear casos de prueba a partir de condiciones de prueba y elementos de cobertura.*
- c) *No es correcto. No es probable que la ejecución de pruebas implique el uso de técnicas de prueba, ya que se ocupa principalmente de ejecutar procedimientos de prueba (y, por tanto, casos de prueba), mientras que las técnicas de prueba crean casos de prueba.*
- d) *No es correcto. No es probable que el seguimiento de las pruebas implique el uso de técnicas de prueba. El monitoreo de pruebas se ocupa principalmente de verificaciones continuas para garantizar que se siga el plan, mientras que las técnicas de prueba crean casos de prueba.*
- e) *Es correcto. Es probable que el análisis de pruebas implique el uso de técnicas de prueba para identificar las condiciones de prueba.*

Mirar sección 2.4.1.

5. Pregunta 5 (1 Punto)

Dado el siguiente software de prueba:

1. Artículos de cobertura.
2. Solicitudes de cambio.
3. Calendario de ejecución de pruebas.
4. Condiciones de prueba priorizadas.

Y las siguientes actividades de prueba.

- A) Análisis de prueba.
- B) Diseño de prueba.
- C) Implementación de pruebas.
- D) Finalización de la prueba.

¿Cuál de las siguientes opciones muestra mejor el software de prueba producido por las actividades?

- a) 1B, 2D, 3C, 4A.
- b) 1B, 2D, 3A, 4C.
- c) 1D, 2C, 3A, 4B.
- d) 1D, 2C, 3B, 4A.

Seleccione una opción.

Solución

Considerando cada una de las actividades de prueba enumeradas y su software de prueba de salida:

- A) *Análisis de prueba: condiciones de prueba priorizadas (4) (por ejemplo, criterios de aceptación) e informes de defectos identificados en la base de prueba.*
- B) *Diseño de pruebas: casos de prueba priorizados, estatutos de prueba, elementos de cobertura (1), requisitos de datos de prueba y requisitos del entorno de prueba.*

- C) *Implementación de pruebas: procedimientos de prueba, scripts de prueba automatizados, conjuntos de pruebas, datos de prueba, cronograma de ejecución de pruebas (3) y elementos del entorno de prueba, como códigos auxiliares, controladores, simuladores y virtualizaciones de servicios.*
- D) *Finalización de la prueba: informe de finalización de la prueba, lecciones aprendidas documentadas, elementos de acción para mejorar y solicitudes de cambio (2) (como elementos de la cartera de productos).*

Así que:

- a) *Es correcta. La coincidencia correcta es: 1B, 2D, 3C, 4A.*
- b) *No es correcta.*
- c) *No es correcta.*
- d) *No es correcta.*

Mirar sección 2.4.1.

6. Pregunta 6 (1 Punto)

¿Cuál de las siguientes afirmaciones sobre las diferentes funciones de prueba es más probable que sea correcta?

- a) En el desarrollo de software ágil, la función de gestión de pruebas es la responsabilidad principal del equipo, mientras que la función de prueba es principalmente responsabilidad de una sola persona externa al equipo.
- b) La función de prueba es la principal responsable del seguimiento y control de las pruebas, mientras que la función de gestión de pruebas es la principal responsable de la planificación y finalización de las pruebas.
- c) En el desarrollo de software ágil, las actividades de gestión de pruebas que abarcan varios equipos son manejadas por un administrador de pruebas externo al equipo, mientras que algunas tareas de gestión de pruebas son manejadas por el propio equipo.
- d) La función de gestión de pruebas es la principal responsable del análisis y el diseño de las pruebas, mientras que la función de pruebas es la principal responsable de la implementación y ejecución de las pruebas.

Seleccione una opción.

Solución

- a) *No es correcto. Aunque es correcto decir que en el desarrollo de software ágil, algunas de las tareas de gestión de pruebas pueden ser manejadas por el propio equipo ágil, la función de prueba no es principalmente responsabilidad de una sola persona externa al equipo. En cambio, es más probable que las pruebas las realicen varios miembros del equipo siguiendo el enfoque de equipo completo.*
- b) *No es correcto. La función de gestión de pruebas implica principalmente actividades relacionadas con la planificación de pruebas, el seguimiento y control de las pruebas y la finalización de las pruebas. Entonces, aunque esta afirmación es parcialmente correcta, es incorrecto decir que la función de prueba es la principal responsable del monitoreo y control de las pruebas.*
- c) *Es correcto. En el desarrollo de software ágil, algunas de las tareas de gestión de pruebas pueden ser manejadas por el propio equipo ágil. Sin embargo, para las actividades de prueba que abarcan varios equipos dentro de una organización, los gerentes de pruebas fuera del equipo de desarrollo pueden realizar estas tareas.*
- d) *No es correcto. La función de gestión de pruebas implica principalmente actividades relacionadas con la planificación de pruebas, el monitoreo y control de pruebas y la finalización de pruebas, mientras que la función de pruebas es principalmente responsable de los aspectos técnicos y de ingeniería de las pruebas, como el análisis de pruebas, el diseño de pruebas, la implementación de pruebas y las pruebas, ejecución. Por lo tanto, la función de gestión de pruebas normalmente no es responsable del análisis y diseño de las pruebas, aunque es correcto decir que la función de pruebas es principalmente responsable de la implementación y ejecución de las pruebas.*

Mirar sección 2.4.5.

7. Pregunta 7 (1 Punto)

¿Cuál de las siguientes es una ventaja del enfoque de todo el equipo?

- a) Equipos sin probadores.
- b) Dinámica de equipo mejorada.
- c) Miembros del equipo de especialistas.
- d) Equipos de mayor tamaño.

Seleccione una opción.

Solución

- a) *No es correcto. En el enfoque de equipo completo, los evaluadores desempeñan un papel vital al compartir su experiencia en pruebas con el equipo y guiar el desarrollo de productos. Colaboran con otros miembros del equipo para lograr los niveles de calidad deseados y trabajan con representantes comerciales para crear pruebas de aceptación. Los evaluadores también se asocian con desarrolladores para determinar la estrategia de prueba óptima y los enfoques de automatización.*
- b) *Es correcto. Al aprovechar las diversas habilidades de cada miembro del equipo de manera más efectiva, el enfoque de todo el equipo fomenta una dinámica de equipo superior, promueve una comunicación y colaboración sólidas y genera un efecto sinérgico que beneficia a todo el proyecto.*
- c) *No es correcto. El enfoque de equipo completo permite que cualquier miembro del equipo con las habilidades y conocimientos necesarios realice cualquier tarea, por lo que los miembros del equipo especializados no son una ventaja de este enfoque.*
- d) *No es correcto. No existe una guía específica sobre el tamaño óptimo de los equipos que utilizan el enfoque de equipo completo, y no hay ninguna sugerencia de que los equipos más grandes sean mejores.*

Mirar sección 2.5.2.

8. Pregunta 8 (1 Punto)

¿Cuál de las siguientes afirmaciones sobre la independencia de las pruebas es correcta?

- a) Los evaluadores independientes encontrarán defectos debido a su perspectiva técnica diferente a la de los desarrolladores, pero su independencia puede conducir a una relación de confrontación con los desarrolladores.
- b) La familiaridad de los desarrolladores con su propio código significa que solo encuentran algunos defectos en él; sin embargo, su experiencia en software compartida con los evaluadores significa que estos defectos también los encontrarían los evaluadores.
- c) Las pruebas independientes requieren evaluadores que estén fuera del equipo del desarrollador e idealmente fuera de la organización; sin embargo, a estos evaluadores les resulta difícil comprender el dominio de la aplicación.
- d) Los evaluadores externos al equipo de desarrolladores son más independientes que los evaluadores internos del equipo, pero es más probable que se culpe a los evaluadores internos del equipo por los retrasos en el lanzamiento del producto.

Seleccione una opción.

Solución

- a) *Es correcto. El principal beneficio de la independencia en las pruebas es que es más probable que los evaluadores identifiquen diferentes tipos de fallas y defectos en comparación con los desarrolladores, debido a sus variados antecedentes, puntos de vista técnicos y posibles sesgos, incluido el sesgo cognitivo. Sin embargo, la principal desventaja de la independencia en las pruebas es que los evaluadores pueden quedar aislados del equipo de desarrollo, lo que genera problemas de comunicación, falta de colaboración, potencialmente, una relación de confrontación, en la que se culpa a los evaluadores de retrasos y cuellos de botella en el proceso de lanzamiento.*
- b) *No es correcto. La familiaridad de un desarrollador con el código no significa que rara vez encuentre defectos en él, sino que esta familiaridad significa que puede encontrar de manera eficiente muchos defectos en su propio código. Y, en lugar de que los desarrolladores y los evaluadores tengan una experiencia compartida, normalmente se cita que los desarrolladores tienen una experiencia diferente a la de los evaluadores como la razón por la que los evaluadores y los desarrolladores encuentran diferentes tipos de defectos.*
- c) *No es correcto. Las pruebas se pueden realizar en diferentes niveles de independencia, que van desde ninguna independencia para el autor hasta una independencia muy alta para los evaluadores externos a la organización. En la mayoría de los proyectos, se utilizan múltiples niveles de independencia: los desarrolladores realizan pruebas de integración y componentes, el equipo de pruebas realiza pruebas de integración y sistemas, y los representantes comerciales realizan pruebas de aceptación. Por lo tanto, los evaluadores pueden estar en el equipo del desarrollador y no es necesario que provengan de fuera de la organización. El conocimiento del dominio de la aplicación cambiará de un caso a otro y no depende del nivel de independencia.*
- d) *No es correcto. Las pruebas se pueden realizar en diferentes niveles de independencia, que van desde ninguna independencia para el autor hasta una independencia muy alta para los evaluadores externos a la organización, siendo los evaluadores externos al equipo del desarrollador generalmente más independientes que los evaluadores internos del equipo. Sin embargo, hay más razones para creer que los evaluadores externos al equipo probablemente estarán más aislados de los desarrolladores y, por lo tanto, es más probable que se les culpe por los retrasos en el lanzamiento del producto.*

Mirar sección 2.5.3.

9. Pregunta 9 (1 Punto)

¿Cuál de las siguientes es una buena práctica de prueba que se aplica a todos los ciclos de vida de desarrollo de software?

- a) Para cada nivel de prueba, existe un nivel de desarrollo correspondiente.
- b) Para cada objetivo de prueba, existe un objetivo de desarrollo correspondiente.
- c) Para cada actividad de prueba de software, existe una actividad de usuario correspondiente.
- d) Para cada actividad de desarrollo de software, existe una actividad de prueba correspondiente.

Seleccione una opción.

Solución

- a) *No es correcto. El control de calidad se aplica a todas las actividades de desarrollo, lo que significa que cada actividad de desarrollo de software tiene una actividad de prueba correspondiente. Sin embargo, aquí intentamos equiparar los niveles de prueba con los niveles de desarrollo y, aunque sabemos lo que se entiende por “niveles de prueba”, no existe una comprensión común del término “nivel de desarrollo”.*
- b) *No es correcto. Cada actividad de desarrollo de software tiene su correspondiente actividad de prueba; sin embargo, los objetivos de las pruebas son bastante diferentes. Por ejemplo, podría haber un objetivo de prueba para garantizar que un objeto de prueba cumpla con un requisito contractual de que se debe realizar un cierto tipo de prueba antes de la entrega. En este caso no hay razón para que exista un objetivo de desarrollo correspondiente.*
- c) *No es correcto. El control de calidad se aplica a todas las actividades de desarrollo, lo que significa que cada actividad de desarrollo de software tiene una actividad de prueba correspondiente. Sin embargo, la misma simetría no se aplica a las pruebas y las actividades de los usuarios. Por ejemplo, para algunos sistemas es difícil incluso identificar a los usuarios finales. Además, algunas actividades de prueba se centran en los desarrolladores (por ejemplo, pruebas de facilidad de mantenimiento), lo que no tiene ningún aspecto de usuario.*
- d) *Es correcto. El control de calidad se aplica a todas las actividades de desarrollo, lo que significa que cada actividad de desarrollo de software tiene una actividad de prueba correspondiente.*

Mirar sección 3.1.2.

10. Pregunta 10 (1 Punto)

¿Cuál de los siguientes es un ejemplo de un enfoque de desarrollo basado en las pruebas primero?

- a) Desarrollo basado en pruebas de componentes.
- b) Desarrollo basado en pruebas de integración.
- c) Desarrollo basado en pruebas del sistema.
- d) Desarrollo basado en pruebas de aceptación.

Seleccione una opción.

Solución

- a) No es correcto. El desarrollo basado en pruebas de componentes no es un ejemplo correcto de un enfoque de desarrollo basado en pruebas.*
- b) No es correcto. El desarrollo basado en pruebas de integración no es un ejemplo correcto de un enfoque de desarrollo basado en pruebas.*
- c) No es correcto. El desarrollo basado en pruebas de sistemas no es un ejemplo correcto de un enfoque de desarrollo basado en pruebas.*
- d) Es correcto. El desarrollo basado en pruebas de aceptación (ATDD) es un ejemplo bien conocido de un enfoque de desarrollo basado en pruebas.*

Mirar sección 3.1.3.

11. Pregunta 11 (1 Punto)

¿Cuál de las siguientes proporciona la mejor descripción del enfoque de desplazamiento a la izquierda?

- a) Cuando lo acuerden los desarrolladores, las actividades manuales en el lado izquierdo del proceso de prueba se automatizan para respaldar el principio de “las pruebas tempranas ahorran tiempo y dinero”.
- b) Cuando sean rentables, las actividades de prueba se trasladan para realizarse más temprano en el ciclo de vida de desarrollo de software (SDLC) para reducir el costo total de calidad al reducir la cantidad de defectos encontrados más adelante en el SDLC.
- c) Cuando tienen tiempo libre disponible, los evaluadores deben automatizar las pruebas para las pruebas de regresión, comenzando con las pruebas de componentes y las pruebas de integración de componentes.

- d) Cuando estén disponibles, los evaluadores están capacitados para realizar tareas en las primeras etapas del SDLC para permitir que se automaticen más actividades de prueba más adelante en el SDLC.

Seleccione una opción.

Solución

- a) *No es correcto. Las prácticas involucradas en las pruebas de desplazamiento a la izquierda tienen como objetivo implementar más actividades de prueba en las primeras fases del ciclo de vida de desarrollo, retratando el SDLC como un movimiento de izquierda a derecha. No existe el lado izquierdo del proceso de prueba.*
- b) *Es correcto. Shift-left o desplazamiento a la izquierda enfatiza la importancia de comenzar las pruebas en una etapa más temprana del ciclo de vida de desarrollo de software (SDLC). La implementación de pruebas de desplazamiento a la izquierda requiere capacitación adicional y un mayor esfuerzo y costos durante las primeras etapas del SDLC; sin embargo, los ahorros generales deberían ser mayores.*
- c) *No es correcto. Aunque las pruebas de componentes automatizadas y las pruebas de integración de componentes para pruebas de regresión son generalmente valiosas, la creación de estas pruebas normalmente es responsabilidad de los desarrolladores, y si se sigue un enfoque de integración continua/ entrega continua (CI/CD), entonces estas pruebas habrán sido enviadas con el código. En algunas situaciones, el evaluador puede automatizar pruebas para pruebas de regresión y, a veces, incluso para pruebas de componentes y pruebas de integración de componentes; sin embargo, esto no es parte de un enfoque de “desplazamiento hacia la izquierda” que mueve las pruebas más temprano en el SDLC.*
- d) *No es correcto. Capacitar a los evaluadores para que realicen tareas en las primeras etapas del SDLC respaldaría un enfoque de desplazamiento a la izquierda al enfatizar la importancia de comenzar las pruebas antes en el SDLC. Sin embargo, automatizar más actividades de prueba que se realizarán más adelante en el SDLC no es parte de un enfoque de “desplazamiento hacia la izquierda”.*

Mirar sección 3.1.5.

12. Pregunta 12 (1 Punto)

¿Cuál de las siguientes situaciones es menos probable que ocurra como resultado de una retrospectiva?

- a) La calidad de los futuros objetos de prueba mejora al identificar mejoras en las prácticas de desarrollo.
- b) La eficiencia de las pruebas mejora al acelerar la configuración de los entornos de prueba a través de la automatización.
- c) Se mejora la comprensión de los usuarios finales sobre los procesos de desarrollo y prueba.
- d) Los scripts de prueba automatizados se mejoran gracias a los comentarios de los desarrolladores.

Seleccione una opción.

Solución

- a) *No es correcto. Uno de los propósitos de las retrospectivas es identificar posibles mejoras en los procesos que, si se ponen en práctica, deberían dar como resultado que la calidad de los resultados futuros del proceso de desarrollo (objetos de prueba) sea mayor. Por lo tanto, es probable que esto ocurra como resultado de una retrospectiva.*
- b) *No es correcto. Un beneficio de las retrospectivas para las pruebas incluye una mayor eficiencia de las pruebas a través de mejoras en los procesos. Por lo tanto, es probable que esto ocurra como resultado de una retrospectiva.*
- c) *Es correcto. Los participantes en las retrospectivas suelen incluir evaluadores, desarrolladores, arquitectos, propietarios de productos y analistas de negocios, pero los usuarios finales rara vez son invitados o asisten a estas reuniones, y es poco probable que reciban informes de estas reuniones. Por lo tanto, es muy poco probable que aprendan y comprendan más sobre los procesos de desarrollo y prueba a través de retrospectivas.*
- d) *No es correcto. Un beneficio de las retrospectivas para las pruebas incluye la mejora de la calidad del software de prueba (incluidos los scripts de prueba automatizados) a través de revisiones conjuntas con los desarrolladores. Por lo tanto, es probable que esto ocurra como resultado de una retrospectiva.*

Mirar sección 3.1.6.

13. Pregunta 13 (1 Punto)

¿Cuál de los siguientes niveles de prueba es más probable que se realice si la prueba se centra en la validación y no la realizan los evaluadores?

- a) Prueba de componentes.
- b) Pruebas de integración de componentes.
- c) Pruebas de integración del sistema.
- d) Pruebas de aceptación.

Seleccione una opción.

Solución

- a) *No es correcto. Las pruebas de componentes (también llamadas pruebas unitarias) implican probar componentes individuales de forma aislada y son principalmente una verificación de una especificación, en lugar de una validación de las necesidades del usuario. Sin embargo, estas pruebas normalmente no las realizan los evaluadores, ya que los desarrolladores suelen realizarlas en su entorno de desarrollo.*
- b) *No es correcto. Las pruebas de integración de componentes implican probar las interfaces y las interacciones entre componentes y son principalmente una verificación de una especificación, en lugar de una validación de las necesidades del usuario. Sin embargo, esta prueba normalmente no la realizan los evaluadores, ya que los desarrolladores suelen realizarla.*
- c) *No es correcto. Las pruebas de integración de sistemas examinan las interfaces con otros sistemas y servicios externos y son principalmente una verificación de una especificación, en lugar de una validación de las necesidades del usuario. Este tipo de prueba también lo realizan con mayor frecuencia los evaluadores.*
- d) *Es correcto. Las pruebas de aceptación se centran en validar que el sistema satisface las necesidades comerciales del usuario y está listo para su implementación. Idealmente, estas pruebas las realizan los usuarios finales.*

Mirar sección 3.2.1.

14. Pregunta 14 (1 Punto)

El software del sistema de navegación se ha actualizado porque sugiere rutas que infringen las leyes de tráfico, como conducir en sentido contrario por calles de sentido único. ¿Cuál de las siguientes describe mejor las pruebas que se realizarán?

- a) Sólo pruebas de confirmación.
- b) Prueba de confirmación y luego prueba de regresión.
- c) Sólo pruebas de regresión.
- d) Pruebas de regresión y luego pruebas de confirmación.

Seleccione una opción.

Solución

- a) *No es correcto. Es necesario realizar pruebas de confirmación para comprobar que las actualizaciones han dado como resultado una implementación correcta; sin embargo, sería sensato realizar pruebas de regresión para garantizar que no se hayan introducido o descubierto defectos en áreas sin cambios del sistema.*
- b) *Es correcto. Las pruebas de confirmación comprobarán que las actualizaciones han dado como resultado una implementación correcta y luego se utilizarán pruebas de regresión para garantizar que no se hayan introducido ni descubierto defectos en áreas sin cambios del sistema.*
- c) *No es correcto. Se deben utilizar pruebas de regresión para garantizar que no se hayan introducido o descubierto defectos en áreas sin cambios del sistema cuando se realizó la actualización; sin embargo, también es necesario realizar pruebas de confirmación que comprobarán que las actualizaciones han dado como resultado una implementación correcta.*
- d) *No es correcto. Las pruebas de confirmación comprobarán que las actualizaciones han dado como resultado una implementación correcta y se utilizarán pruebas de regresión para garantizar que no se hayan introducido ni descubierto defectos en áreas sin cambios del sistema. Sin embargo, cuando se realiza (es decir, cuando es necesario probar una actualización), la prueba de confirmación precede a la prueba de regresión.*

Mirar sección 3.2.3.

15. Pregunta 15 (1 Punto)

Dados los siguientes defectos de ejemplo:

- I. Dos partes diferentes de la especificación de diseño no están de acuerdo debido a la complejidad del diseño.
- II. El tiempo de respuesta es demasiado largo y hace que los usuarios pierdan la paciencia.
- III. No se puede alcanzar una ruta en el código durante la ejecución.
- IV. Una variable se declara pero nunca se utiliza posteriormente en el programa.
- V. La cantidad de memoria que necesita el programa para generar un informe es demasiado alta.

¿Cuál de los siguientes identifica mejores ejemplos de defectos que podrían encontrarse mediante pruebas estáticas (en lugar de pruebas dinámicas)?

- a) II, V.
- b) III, V.
- c) I, II, IV.
- d) I, III, IV.

Seleccione una opción.

Solución

Considerando cada uno de los ejemplos de defectos enumerados:

- I. Dos partes diferentes de la especificación de diseño no están de acuerdo debido a la complejidad del diseño; este es un ejemplo de un defecto de especificación, que incluye inconsistencias, ambigüedades, contradicciones, omisiones, imprecisiones y duplicaciones, que se pueden encontrar más fácilmente mediante pruebas estáticas.*
- II. Un tiempo de respuesta es demasiado largo y, por lo tanto, hace que los usuarios pierdan la paciencia; este es un ejemplo de un defecto en el tiempo de respuesta, que en la práctica sólo puede detectarse ejecutando el programa y midiendo el tiempo de respuesta, que se puede encontrar más fácilmente mediante pruebas dinámicas.*

- III. *No se puede alcanzar una ruta en el código durante la ejecución: este es un ejemplo de un defecto de codificación, que incluye variables con valores indefinidos, variables no declaradas, código duplicado o inalcanzable y complejidad excesiva del código, que se puede encontrar más fácilmente mediante pruebas estáticas.*
- IV. *Una variable se declara, pero nunca se utiliza posteriormente en el programa; este es un ejemplo de un defecto de codificación, que incluye variables con valores indefinidos, variables no declaradas, código duplicado o inalcanzable y una complejidad excesiva del código, que se puede encontrar más fácilmente mediante pruebas estáticas.*
- V. *La cantidad de memoria que necesita el programa para generar un informe es demasiado alta; este es un ejemplo de un defecto de rendimiento, que sólo puede detectarse en la práctica ejecutando el programa y midiendo la memoria utilizada, que se puede encontrar más fácilmente. Mediante pruebas dinámicas.*

Así que:

- a) *No es correcto.*
- b) *No es correcto.*
- c) *No es correcto.*
- d) *Es correcto. La coincidencia correcta para las pruebas estáticas es I, III y IV.*

Mirar sección 4.1.3.

16. Pregunta 16 (1 Punto)

¿Cuál de los siguientes es un beneficio de la retroalimentación temprana y frecuente de las partes interesadas?

- a) Los cambios en los requisitos se entienden e implementan antes.
- b) Garantiza que las partes interesadas del negocio comprendan los requisitos de los usuarios.
- c) Permite a los propietarios de productos cambiar sus requisitos con la frecuencia que quieran.
- d) Se informa a los usuarios finales qué requisitos no se implementarán antes del lanzamiento.

Seleccione una opción.

Solución

- a) *Es correcto. Obtener comentarios de las partes interesadas desde el principio y con frecuencia durante el proceso de desarrollo de software puede resultar muy beneficioso. Facilita la comunicación temprana de posibles problemas de calidad, puede evitar malentendidos sobre los requisitos y garantiza que cualquier cambio en los requisitos de las partes interesadas se comprenda e implemente antes.*
- b) *No es correcto. Los comentarios provienen de las partes interesadas y es poco probable que su aportación mejore su comprensión de los requisitos de sus propios usuarios.*
- c) *No es correcto. Obtener comentarios de las partes interesadas desde el principio y con frecuencia durante el proceso de desarrollo de software puede resultar muy beneficioso. Facilita la comunicación temprana de posibles problemas de calidad, puede evitar malentendidos sobre los requisitos y garantiza que cualquier cambio en los requisitos de las partes interesadas se comprenda e implemente antes. Sin embargo, el hecho de que los cambios en los requisitos puedan entenderse e implementarse antes no significa que se fomenten cambios ilimitados en los requisitos.*
- d) *No es correcto. La retroalimentación proviene de las partes interesadas y no cubre la comunicación con ellas. Las comunicaciones a los usuarios finales podrían incluir informarles sobre qué requisitos no se implementarán antes del lanzamiento, pero idealmente esto no debería suceder en absoluto.*

Mirar sección 4.2.1.

17. Pregunta 17 (1 Punto)

Dados los siguientes tipos de revisión:

1. Revisión técnica.
2. Revisión informal.
3. Inspección.
4. Tutorial.

Y las siguientes descripciones:

- A) Incluye objetivos como lograr consenso, generar nuevas ideas y motivar a los autores a mejorar.
- B) Incluye objetivos como educar a los revisores, lograr consenso, generar nuevas ideas y detectar defectos potenciales.
- C) El objetivo principal es detectar defectos potenciales y requiere la recopilación de métricas para respaldar la mejora del proceso.
- D) El objetivo principal es detectar defectos potenciales y no genera resultados documentados formales.

¿Cuál de las siguientes opciones coincide mejor con los tipos de reseñas y las descripciones?

- a) 1A, 2B, 3C, 4D.
- b) 1A, 2D, 3C, 4B.
- c) 1B, 2C, 3D, 4A.
- d) 1C, 2D, 3A, 4B.

Seleccione una opción.

Solución

Considerando cada uno de los tipos de revisión enumerados:

1. *Revisión técnica: este tipo de revisión la realizan revisores técnicamente calificados y está dirigida por un moderador. Los objetivos son lograr consenso y tomar decisiones sobre problemas técnicos, al mismo tiempo que se evalúa la calidad y se genera confianza en el producto del trabajo, se generan nuevas ideas, se motiva y capacita a los autores para mejorar y se detectan anomalías.*
2. *Revisión informal: el objetivo principal es detectar anomalías. El proceso no está definido y no requiere resultados documentados formales.*
3. *Inspección: este es el tipo de revisión más formal y sigue el proceso de revisión genérico completo. El objetivo principal es encontrar la mayor cantidad de anomalías, y otros objetivos incluyen evaluar la calidad y generar confianza en el producto del trabajo, motivar y permitir a los autores mejorar y recopilar métricas que puedan usarse para mejorar el*

ciclo de vida de desarrollo de software (SDLC), incluyendo el proceso de inspección. El autor no puede actuar como líder de revisión o escriba.

4. Tutorial: dirigido por el autor, este tipo de revisión cumple varios objetivos, como evaluar la calidad y generar confianza en el producto del trabajo, educar a los revisores, obtener consenso, generar nuevas ideas, motivar y permitir a los autores mejorar y detectar anomalías. Los revisores pueden realizar una revisión individual antes del recorrido, pero esto no es obligatorio.

- A) Incluye objetivos como lograr consenso, generar nuevas ideas y motivar a los autores a mejorar (A).*
- B) El objetivo principal es detectar defectos potenciales y no genera resultados documentados formales (D).*
- C) El objetivo principal es detectar defectos potenciales y requiere la recopilación de métricas para respaldar la mejora del proceso (C).*
- D) Incluye objetivos como educar a los revisores, lograr consenso, generar nuevas ideas y detectar posibles defectos (B).*

Así que:

- a) No es correcto.*
- b) Es correcto. La coincidencia correcta es: 1A, 2D, 3C, 4B.*
- c) No es correcta.*
- d) No es correcta.*

Mirar sección 3.2.4.

18. Pregunta 18 (1 Punto)

¿Cuál de los siguientes es un factor que contribuye a una revisión exitosa?

- a) Garantizar que la dirección participe como revisores.
- b) Dividir los productos de trabajo grandes en partes más pequeñas.
- c) Establecer como objetivo la evaluación del revisor.
- d) Planee cubrir un documento por revisión.

Seleccione una opción.

Solución

- a) *No es correcto. Para garantizar revisiones exitosas, es importante asegurar el apoyo de la administración para el proceso de revisión; sin embargo, eso no significa que deban participar como revisores.*
- b) *Es correcto. Para garantizar revisiones exitosas, es importante dividir el producto de trabajo en partes que sean lo suficientemente pequeñas como para ser revisadas en un plazo de tiempo razonable para evitar que los revisores pierdan el enfoque durante la revisión individual o las reuniones de revisión.*
- c) *No es correcto. Para garantizar revisiones exitosas, es importante definir claramente objetivos y criterios de salida mensurables, sin evaluar a los participantes.*
- d) *No es correcto. Para garantizar revisiones exitosas, es importante dividir la revisión en partes más pequeñas para evitar que los revisores pierdan el enfoque durante la revisión individual o las reuniones de revisión. Por lo tanto, no debes planear cubrir un documento por revisión.*

Mirar sección 4.2.5.

19. Pregunta 19 (1 Punto)

¿Cuál es la principal diferencia entre las técnicas de prueba de caja negra y las técnicas de prueba basadas en la experiencia?

- a) El objeto de prueba.
- b) El nivel de prueba en el que se utiliza la técnica de prueba.
- c) La base de la prueba.
- d) El ciclo de vida de desarrollo de software (SDLC) en el que se puede utilizar la técnica de prueba.

Seleccione una opción.

Solución

- a) *No es correcto. En la mayoría de los casos, se pueden utilizar tanto técnicas de prueba de caja negra como técnicas de prueba basadas en la experiencia para los mismos objetos de prueba.*

- b) *No es correcto. Tanto las técnicas de prueba de caja negra como las técnicas de prueba basadas en la experiencia se pueden utilizar en todos los niveles de prueba.*
- c) *Es correcto. Las técnicas de prueba de caja negra (también conocidas como técnicas basadas en especificaciones) se basan en un análisis del comportamiento especificado del objeto de prueba sin referencia a su estructura interna. Por tanto, la base de la prueba suele ser una especificación. Las técnicas de prueba basadas en la experiencia utilizan eficazmente el conocimiento y la experiencia de los evaluadores para el diseño y la implementación de casos de prueba. Esto significa que el evaluador, al diseñar pruebas, puede no utilizar la especificación en absoluto.*
- d) *No es correcto. Las técnicas de prueba basadas en la experiencia pueden detectar defectos que pueden pasarse por alto utilizando técnicas de prueba de caja negra (y caja blanca). Por lo tanto, las técnicas de prueba basadas en la experiencia son complementarias de las técnicas de prueba de caja negra y de la caja blanca, y tanto las técnicas de prueba de caja negra como las técnicas de prueba basadas en la experiencia se pueden utilizar en todos los SDLC.*

Mirar sección 5.1.1.

20. Pregunta 20 (1 Punto)

Está probando un validador de PIN, que acepta PIN válidos y rechaza PIN no válidos. Un PIN es una secuencia de dígitos. Un PIN es válido si consta de cuatro dígitos, de los cuales al menos dos son diferentes. Ha identificado las siguientes particiones de equivalencia válidas:

- Variable: longitud del código PIN.
 - La partición de “longitud correcta”: PIN de cuatro dígitos.
 - La partición tiene “longitud incorrecta”: PIN con una longitud distinta de 4.
- Variable: número de dígitos diferentes.
 - La partición “número de dígitos diferentes correcto”: PIN con al menos dos dígitos diferentes.
 - La partición “número de dígitos diferentes incorrecto”: PIN con todos los dígitos iguales.

¿Cuál de los siguientes es un conjunto mínimo de datos de prueba de entrada que cubre todas las particiones de equivalencia identificadas?

- a) 1234, 12345, 1.
- b) 111, 1111, 1112.
- c) 1,1234.
- d) 12345,1234.

Seleccione una opción.

Solución

- a) *No es correcto. Estos tres valores logran una cobertura total de las particiones de equivalencia, pero se puede lograr una cobertura total con dos valores.*
- b) *No es correcto. Estos tres valores logran una cobertura total de las particiones de equivalencia, pero se puede lograr una cobertura total con dos valores.*
- c) *Es correcto. El valor “1” cubre “longitud incorrecta” y “número de dígitos diferentes incorrectos”. El valor “1234” cubre “longitud correcta” y “número de dígitos diferentes correctos”. Estos dos valores cubren las cuatro particiones de equivalencia identificadas.*
- d) *No es correcto. Este conjunto no cubre la partición de equivalencia “número de dígitos diferentes incorrectos”.*

Mirar sección 5.2.1.

21. Pregunta 21 (1 Punto)

W se pidió a un desarrollador que implementara la siguiente regla comercial:

INPUT: value (integer number).

IF (value \leq 100 OR value \geq 200) THEN write “value incorrect”.

ELSE write “value OK”.

Los casos de prueba se diseñan utilizando el análisis de valores límite de 2 valores.

¿Cuál de los siguientes conjuntos de entradas de prueba logra la mayor cobertura?

- a) 100, 150, 200, 201.
- b) 99, 100, 200, 201.
- c) 98, 99, 100, 101.
- d) 101, 150, 199, 200.

Seleccione una opción.

Solución

Las particiones de equivalencia son: {..., 99, 100}, {101, 102..., 198, 199}, {200, 201...}. Por lo tanto, hay 4 valores límite, que son: 100, 101, 199 y 200. En BVA de 2 valores, para cada valor límite hay dos elementos de cobertura (el valor límite y su vecino más cercano que pertenece a la partición adyacente). Como los vecinos más cercanos también son valores límite en la partición adyacente, solo hay cuatro elementos de cobertura.

Así que:

- a) No es correcta. Sólo 100 y 200 son artículos de cobertura válidos para BVA de 2 valores, por lo que logramos una cobertura del 50%.
- b) No es correcto. Sólo 100 y 200 son artículos de cobertura válidos para BVA de 2 valores, por lo que logramos una cobertura del 50%.
- c) No es correcto. Sólo 100 y 101 son artículos de cobertura válidos para BVA de 2 valores, por lo que logramos una cobertura del 50%.
- d) Es correcto. 101, 199 y 200 son artículos de cobertura válidos para BVA de 2 valores, por lo que logramos una cobertura del 75%.

Mirar sección 5.2.2.

22. Pregunta 22 (1 Punto)

Está trabajando en un proyecto para desarrollar un sistema para analizar los resultados de los exámenes de conducción. Se le ha pedido que diseñe casos de prueba basados en la siguiente tabla de decisiones.

	R1	R2	R3
C1: ¿Primer intento en el examen?	-	-	F
C2: ¿Examen teórico pasado?	T	F	-
C3: ¿Examen práctico pasado?	T	-	F
Expedir licencia de conducir	X		
¿Solicitudes adicionales de lecciones de conducir?			X
¿Solicitud para tomar el examen de nuevo?		X	

¿Qué datos de prueba mostrarán que hay reglas contradictorias en la tabla de decisiones?

- a) C1 = T, C2 = T, C3 = F
- b) C1 = T, C2 = F, C3 = T
- c) C1 = T, C2 = T, C3 = T y C1 = F, C2 = T, C3 = T
- d) C1 = F, C2 = F, C3 = F

Seleccione una opción.

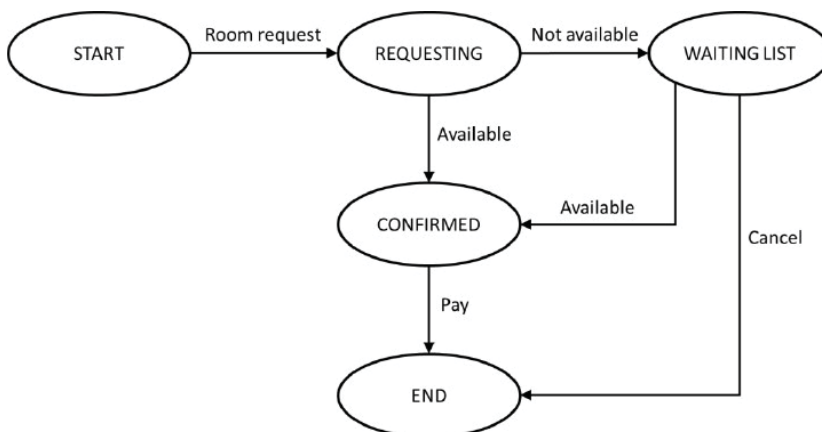
Solución

- a) No es correcto. La combinación (T, T, F) no coincide con ninguna regla. Este es un ejemplo de omisión, no una contradicción.
- b) No es correcto. La combinación (T, F, T) coincide solo con una columna, R2, por lo que no hay contradicción.
- c) No es correcto. Ambas combinaciones (T, T, T) y (F, T, T) coinciden solo con una columna, R1, por lo que no hay contradicción.
- d) *Es correcto. La combinación (F, F, F) coincide con R2 y R3, pero R2 y R3 tienen acciones diferentes, por lo que esto muestra una contradicción entre R2 y R3.*

Mirar sección 5.2.3.

23. Pregunta 23 (1 Punto)

Está diseñando casos de prueba basados en el siguiente diagrama de transición de estado:



¿Cuál es el número mínimo de casos de prueba necesarios para lograr una cobertura de transiciones válida al 100 % ?

- a) 3
- b) 2
- c) 5
- d) 6

Seleccione una opción.

Solución

Las siguientes tres transiciones:

*“SOLICITANDO -> CONFIRMADO” “LISTA DE ESPERA -> CONFIRMADO”
“LISTA DE ESPERA -> FINALIZAR”.*

No puede aparecer en el mismo caso de prueba, lo que sugiere que se requieren al menos tres casos de prueba. Todas las demás transiciones pueden aparecer en combinación con una o más de estas tres transiciones, por lo que necesitamos un mínimo de tres casos de prueba. De hecho, sólo son posibles tres secuencias:

▼ CP1:

- *INICIO (Solicitud de habitación).*
- *SOLICITANDO (Disponible).*
- *CONFIRMADO (Pago.).*
- *FIN.*

▼ CP2:

- *INICIO (Solicitud de habitación).*
- *SOLICITANDO (No disponible).*
- *LISTA DE ESPERA (Disponible).*
- *CONFIRMADO (Pagar).*
- *FIN.*

▼ TC3:

- *INICIO (Solicitud de habitación).*
- *SOLICITANDO (No disponible).*
- *LISTA DE ESPERA (Cancelar).*
- *FIN.*

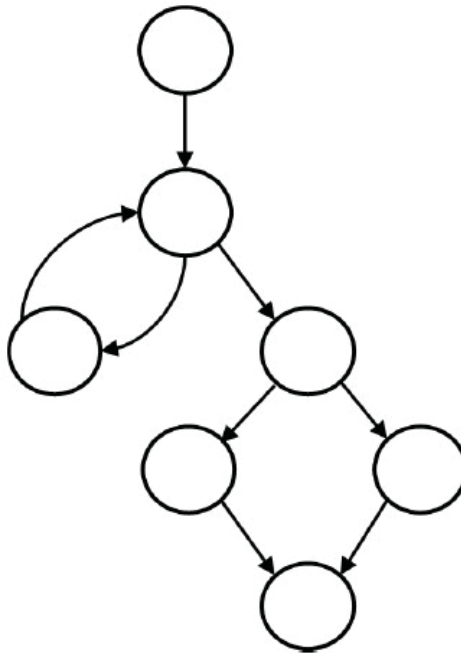
Así que:

- a) *Es correcto.*
- b) *No es correcto.*
- c) *No es correcto.*
- d) *No es correcto.*

Mirar sección 5.2.4.

24. Pregunta 24 (1 Punto)

Desea aplicar pruebas de rama al código representado por el siguiente gráfico de flujo de control.



¿Cuántos elementos de cobertura necesita probar?

- a) 2
- b) 4
- c) 8
- d) 7

Seleccione una opción.

Solución

En las pruebas de rama, los elementos de cobertura son ramas, que están representados por los bordes de un gráfico de flujo de control. Hay 8 aristas en el gráfico de flujo de control.

Así que:

- a) No es correcto.*
- b) No es correcto.*
- c) Es correcto.*
- d) No es correcto.*

Mirar sección 5.3.2.

25. Pregunta 25 (1 Punto)

¿Cómo pueden ser útiles las pruebas de caja blanca para respaldar las pruebas de caja negra?

- a) Las medidas de cobertura de caja blanca pueden ayudar a los evaluadores a evaluar las pruebas de caja negra en términos de la cobertura de código lograda por estas pruebas de caja negra.
- b) El análisis de cobertura de caja blanca puede ayudar a los evaluadores a identificar fragmentos inalcanzables del código fuente.
- c) Las pruebas de sucursal incluyen técnicas de prueba de caja negra, por lo que lograr una cobertura total de la sucursal garantiza lograr una cobertura total de cualquier técnica de caja negra.
- d) Las técnicas de prueba de caja blanca pueden proporcionar elementos de cobertura para las técnicas de caja negra.

Seleccione una opción.

Solución

- a) Es correcto. Realizar únicamente pruebas de caja negra no proporciona una medida de la cobertura real del código. Las medidas de cobertura de caja blanca proporcionan una medición objetiva de la cobertura y proporcionan la información necesaria para permitir que se generen pruebas adicionales para aumentar esta cobertura y, posteriormente, aumentar la confianza en el código.*

- b) No es correcto. Esta afirmación es falsa y además no responde la pregunta (no tiene nada que ver con las pruebas de caja negra).*
- c) No es correcto. En general, no existen relaciones subsume entre las técnicas de caja blanca y de caja negra.*
- d) No es correcto. Las técnicas de caja blanca se utilizan para diseñar pruebas basadas en el propio objeto de prueba, mientras que las técnicas de caja negra se utilizan para diseñar pruebas basadas en la especificación. Por tanto, no existe relación entre las partidas de cobertura derivadas de estos dos tipos de técnicas.*

Mirar sección 5.3.3.

26. Pregunta 26 (1 Punto)

Considere la siguiente lista:

- Entrada correcta no aceptada.
- Se aceptan entradas incorrectas.
- Formato de salida incorrecto.
- División por cero.

¿Qué técnica de prueba utiliza más probablemente el evaluador que utiliza esta lista al realizar la prueba?

- a) Pruebas exploratorias.
- b) Ataque de falla.
- c) Pruebas basadas en listas de verificación.
- d) Análisis de valor límite.

Seleccione una opción.

Solución

- a) No es correcto. Las pruebas exploratorias utilizan cartas de prueba, no una lista de posibles defectos/fallas. Aunque las pruebas exploratorias pueden incorporar el uso de otras técnicas de prueba, en este caso el ataque a fallos es la opción más probable.*
- b) Es correcto. Esta es una lista de posibles fallos. Los ataques a fallas son un enfoque metódico para la implementación de la adivinación de errores y requieren que el evaluador cree o adquiera una lista de posibles errores,*

defectos y fallas, y que diseñe pruebas que identifiquen defectos asociados con los errores, expongan los defectos o causen el error. Fracasos.

- c) *No es correcto. El evaluador utiliza una lista de verificación de elementos para respaldar sus pruebas. Tanto la adivinación de errores como las pruebas basadas en listas de verificación utilizan dichas listas; sin embargo, la lista aquí es de posibles fallas, no de condiciones de prueba, por lo que la técnica de prueba más probable es el ataque de fallas, que se enfoca en errores, defectos y fallas.*
- d) *No es correcto. BVA se basa en un análisis de los valores límite de las particiones de equivalencia. La lista anterior no menciona las particiones de equivalencia ni sus límites.*

Mirar sección 5.4.1.

27. Pregunta 27 (1 Punto)

¿Cuál de las siguientes opciones describe mejor cómo el uso de pruebas basadas en listas de verificación puede dar como resultado una mayor cobertura?

- a) Los elementos de la lista de verificación se pueden definir con un nivel de detalle suficientemente bajo, de modo que el evaluador pueda implementar y ejecutar casos de prueba detallados basados en estos elementos.
- b) Las listas de verificación se pueden automatizar, de modo que cada vez que la ejecución de una prueba automatizada cubra los elementos de la lista de verificación, se obtendrá una cobertura adicional.
- c) Cada elemento de la lista de verificación debe probarse por separado e independientemente, de modo que los elementos cubran diferentes áreas del software.
- d) Dos evaluadores que diseñan y ejecutan pruebas basadas en los mismos elementos de la lista de verificación de alto nivel normalmente realizarán las pruebas de maneras ligeramente diferentes.

Seleccione una opción.

Solución

- a) *No es correcto. Si bien es cierto que el evaluador puede implementar y ejecutar casos de prueba detallados basados en la lista de verificación, no explica cómo esto resultaría en una mayor cobertura.*

- b) No es correcto. Los elementos de la lista de verificación no deben automatizarse. Pero incluso si lo son, los scripts de prueba automatizados siempre ejecutan las pruebas de la misma manera, lo que generalmente no resulta en una mayor cobertura.*
- c) No es correcto. Es cierto que cada elemento de la lista de verificación debe probarse por separado e independientemente. Pero esto afecta el orden de ejecución de la prueba y no afecta la cobertura lograda, por lo que no resulta en una mayor cobertura.*
- d) Es correcto. Si las listas de verificación son de alto nivel, es probable que se produzca cierta variabilidad en las pruebas reales, lo que dará lugar a una cobertura potencialmente mayor pero a una menor repetibilidad. Si dos evaluadores siguen una lista de verificación de elementos de alto nivel, cada uno de ellos puede usar diferentes datos de prueba, pasos de prueba, etc. De esta manera, un evaluador probablemente cubrirá algunas áreas no cubiertas por el otro evaluador y esto dará como resultado una mayor cobertura.*

Mirar sección 5.4.3.

28. Pregunta 28 (1 Punto)

¿Cuál de los siguientes proporciona el mejor ejemplo de un criterio de aceptación orientado a escenarios?

- a) La aplicación debe permitir a los usuarios eliminar su cuenta y todos los datos asociados previa solicitud.
- b) Cuando un cliente agrega un artículo a su carrito y procede al pago, se le debe solicitar que inicie sesión o cree una cuenta si aún no lo ha hecho.
- c) SI (contiene (producto (23). Nombre, carrito. Productos ())) Entonces devuelve Falso.
- d) El sitio web debe cumplir con los Estándares 508 de Accesibilidad a las TIC y garantizar que todo el contenido sea accesible para usuarios con discapacidades.

Seleccione una opción.

Solución

- a) *No es correcto. Este criterio de aceptación describe qué reglas o regulaciones debe cumplir el sistema (en este caso, el derecho al olvido). Este es un ejemplo de un criterio de aceptación orientado a reglas.*
- b) *Es correcto. Este criterio de aceptación describe un escenario de ejemplo que debe ser realizable por el sistema. Este es un ejemplo de un criterio de aceptación orientado a escenarios.*
- c) *No es correcto. Esta oración se parece más a una línea de código que implementa alguna regla comercial. Los criterios de aceptación deben redactarse en colaboración con los representantes comerciales y, por lo tanto, deben redactarse en un lenguaje que comprendan. Lo más probable es que esta frase resulte ininteligible para estas partes interesadas.*
- d) *No es correcto. Este criterio de aceptación describe qué reglas o regulaciones debe cumplir el sistema y cómo se garantizará el cumplimiento. Por lo tanto, este es un ejemplo de un criterio de aceptación orientado a reglas, no un criterio de aceptación basado en escenarios.*

Mirar sección 5.5.2.

29. Pregunta 29 (1 Punto)

Está utilizando el desarrollo basado en pruebas de aceptación y diseñando casos de prueba basados en la siguiente historia de usuario:

Como usuario Regular o Especial, quiero poder utilizar mi tarjeta de piso electrónica, para acceder a pisos específicos.

Criterios de aceptación:

- AC1: los usuarios habituales tienen acceso a los pisos 1 al 3.
- AC2: el piso 4 solo es accesible para usuarios especiales.
- AC3: los usuarios especiales tienen todos los derechos de acceso de los usuarios regulares.

¿Qué caso de prueba es el más razonable para probar AC3?

- a) Comprobar que un usuario regular puede acceder a los pisos 1 y 3.
- b) Comprobar que un usuario regular no pueda acceder al piso 4.
- c) Comprobar que un usuario especial puede acceder al piso 5.
- d) Comprobar que un usuario especial puede acceder a los pisos 1, 2 y 3.

Seleccione una opción.

Solución

- a) *No es correcto. Queremos verificar que los usuarios especiales tengan los derechos de los usuarios regulares, por lo que debemos probar los derechos de acceso para un usuario especial, no para un usuario regular.*
- b) *No es correcto. Queremos verificar que los usuarios especiales tengan los derechos de los usuarios regulares, por lo que debemos probar los derechos de acceso para un usuario especial, no para un usuario regular.*
- c) *No es correcto. No existe ningún piso 5 descrito en los criterios de aceptación. Los casos de prueba no deben ampliar el alcance de la historia del usuario. Pero incluso si quisiéramos realizar una prueba negativa, esta prueba no está directamente relacionada con AC3.*
- d) *Es correcto. De esta manera podemos comprobar si un usuario especial puede acceder a pisos a los que puede acceder un usuario regular.*

Mirar sección 5.5.3.

30. Pregunta 30 (1 Punto)

¿Cuál de los siguientes no es un propósito de un plan de prueba?

- a) Definir datos de prueba y resultados esperados para pruebas de componentes y pruebas de integración de componentes.
- b) Definir como criterio de salida del nivel de prueba de componentes que “se debe alcanzar el 100% de cobertura del estado de cuenta y el 100% de cobertura de sucursales”.
- c) Describir qué campos debe contener el informe de progreso de la prueba y cuál debe ser la forma de este informe.
- d) Explicar por qué las pruebas de integración del sistema se excluirán de las pruebas, aunque la estrategia de prueba requiere este nivel de prueba.

Seleccione una opción.

Solución

- a) *Es correcto. El plan de prueba puede incluir requisitos de datos de prueba (como parte del enfoque de prueba), pero no los datos de prueba detallados para los casos de prueba. Los datos de prueba son parte de los casos de prueba, no del plan de prueba. Además, normalmente es imposible definir dichos datos cuando se crea el plan de prueba, porque no se sabe exactamente cómo serán los componentes.*

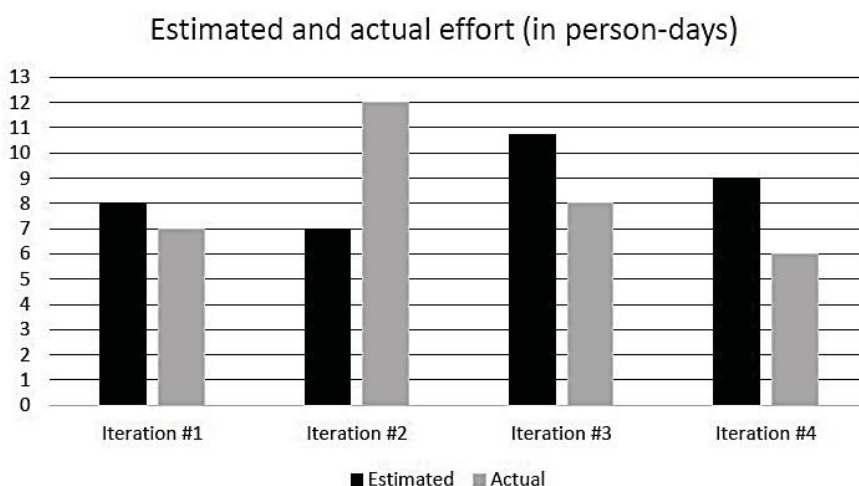
- b) No es correcto. Uno de los propósitos de un plan de prueba es ayudar a garantizar que las actividades de prueba realizadas cumplan con los criterios establecidos, al incluir criterios de entrada y criterios de salida. Los criterios de cobertura del código son un ejemplo de dichos criterios para el nivel de prueba de componentes.
- c) No es correcto. Las plantillas de documentación son el contenido típico de un plan de prueba. Esto ayuda a facilitar la comunicación entre las partes interesadas al definir una forma estándar de comunicación o presentación de informes.
- d) No es correcto. Uno de los propósitos de un plan de pruebas es demostrar que las pruebas se adherirán a la política y estrategia de pruebas existentes, o explicar por qué las pruebas se desviarán de ellas. Este es un ejemplo de cómo explicar la desviación, con respecto a los niveles de prueba que se seguirán (o no).

Mirar sección 6.1.1.

31. Pregunta 31 (1 Punto)

Al comienzo de cada iteración, el equipo estima la cantidad de trabajo (en días-persona) que necesitarán completar durante la iteración. Sea $E(n)$ la cantidad de trabajo estimada para la iteración n , y sea $A(n)$ la cantidad real de trabajo realizado en la iteración n . A partir de la tercera iteración, el equipo utiliza el siguiente modelo de estimación basado en extrapolación: $(n)=3*A(n-1) + A(n-2)4$

El gráfico muestra la cantidad de trabajo estimada y real para las primeras cuatro iteraciones.



¿Cuál es la cantidad estimada de trabajo para la iteración N° 5?

- a) 10,5 días-persona.
- b) 8,25 días-persona.
- c) 6,5 días-persona.
- d) 9,4 días-persona.**

Seleccione una opción.

Solución

Del gráfico tenemos: $A(4) = 6$ y $A(3) = 8$ (los dos últimos cuadros grises).

*De la fórmula obtenemos: $E(5) = (3 * A(4) + A(3)) / 4 = (3 * 6 + 8) / 4 = 26 / 4 = 6,5$ días-persona.*

Así que:

- a) No es correcto.*
- b) No es correcto.*
- c) Es correcto.*
- d) No es correcto.*

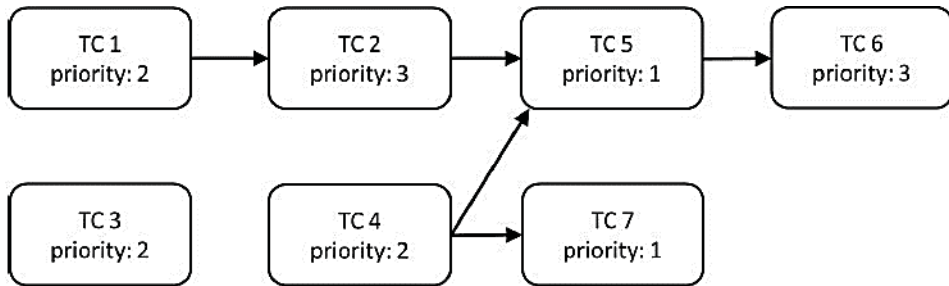
Mirar la sección 6.1.4

32. Pregunta 32 (1 Punto)

Está preparando un programa de ejecución de pruebas para ejecutar siete casos de prueba TC 1 a TC 7.

La siguiente figura incluye las prioridades de estos casos de prueba (1 = prioridad más alta, 3 = prioridad más baja).

La figura también muestra las dependencias entre casos de prueba mediante flechas. Por ejemplo, la flecha de TC 4 a TC 5 significa que TC 5 solo se puede ejecutar si TC 4 se ejecutó previamente.



¿Qué caso de prueba debería ejecutarse en sexto lugar?

- a) CT 3.
- b) CT 5.
- c) CT 6.
- d) CT 2.

Seleccione una opción.

Solución

Queremos ejecutar casos de prueba según sus prioridades, pero también debemos considerar las dependencias.

Si solo consideramos las prioridades, primero queremos ejecutar TC 5 y TC 7 (prioridad más alta), luego TC 1, TC 3 y TC 4, y finalmente TC 2 y TC 6 (prioridad más baja).

Sin embargo, para ejecutar TC 7, primero debemos ejecutar TC 4. Para ejecutar TC 5, necesitamos ejecutar TC 4 y TC 2, pero TC 2 está bloqueado por TC 1, que debe ejecutarse antes de TC 2.

Entonces, para ejecutar casos de prueba de prioridad 1 lo antes posible, los primeros cinco casos de prueba deben ser: TC 4–TC 7–TC 1–TC 2–TC 5.

A continuación, necesitamos ejecutar TC 3, porque tiene mayor prioridad que TC 6. Por lo tanto, el cronograma completo será TC 4 – TC 7 – TC 1 – TC 2 – TC 5 – TC 3 – TC 6.

Por lo tanto, el sexto caso de prueba será TC 3.

Así que:

- a) *Es correcto.*
- b) *No es correcto.*
- c) *No es correcto.*
- d) *No es correcto.*

33. Pregunta 33 (1 Punto)

¿Qué muestra el modelo piramidal de prueba?

- a) Que las pruebas pueden tener diferentes prioridades.
- b) Que las pruebas pueden tener diferente granularidad.
- c) Que las pruebas pueden requerir criterios de cobertura diferentes.
- d) Que las pruebas pueden depender de otras pruebas.

Seleccione una opción.

Solución

- a) *No es correcto. El modelo de pirámide de pruebas no proporciona información sobre las prioridades de las pruebas.*
- b) *Es correcto. El modelo de pirámide de pruebas muestra que diferentes pruebas tienen diferentes niveles de granularidad.*
- c) *No es correcto. El modelo piramidal de prueba es independiente de los criterios de cobertura.*
- d) *No es correcto. El modelo de pirámide de pruebas no muestra ninguna relación entre diferentes pruebas.*

Mirar sección 6.1.6.

34. Pregunta 34 (1 Punto)

¿Cuál es la relación entre los cuadrantes de prueba, los niveles de prueba y los tipos de prueba?

- a) Los cuadrantes de prueba representan combinaciones particulares de niveles y tipos de prueba, definiendo su ubicación en el ciclo de vida del desarrollo de software.
- b) Los cuadrantes de prueba describen el grado de granularidad de los tipos de prueba individuales realizados en cada nivel de prueba.

- c) Los cuadrantes de prueba asignan los tipos de prueba que se pueden realizar a los niveles de prueba.
- d) Los cuadrantes de prueba agrupan niveles de prueba y tipos de prueba según varios criterios, como dirigirse a partes interesadas específicas.

Solución

- a) *No es correcto. Los cuadrantes de prueba agrupan los niveles y tipos de prueba por separado según varios criterios. No representan ninguna combinación de niveles y tipos de prueba y no están relacionados con ninguna ubicación dentro del ciclo de vida de desarrollo de software. Tanto los niveles como los tipos de prueba se tratan por separado en el modelo de cuadrantes de prueba.*
- b) *No es correcto. Los cuadrantes de prueba agrupan niveles y tipos de pruebas según varios criterios. No describen el grado de granularidad de los tipos de pruebas individuales realizados en cada nivel de prueba. Este modelo, en cuanto a los niveles de prueba, se llama pirámide de prueba.*
- c) *No es correcto. La afirmación es incorrecta porque, en general, cualquier tipo de prueba se puede realizar en cualquier nivel de prueba.*
- d) *Es correcto. Los cuadrantes de prueba agrupan niveles de prueba, tipos de prueba, actividades, técnicas de prueba y productos de trabajo en el desarrollo de software ágil. En este modelo, las pruebas pueden estar orientadas al negocio o a la tecnología. Las pruebas pueden apoyar al equipo (es decir, guiar el desarrollo) o criticar el producto (es decir, medir su comportamiento frente a las expectativas). La combinación de estos dos puntos de vista determina los cuatro cuadrantes.*

Mirar sección 6.1.7.

35. Pregunta 35 (1 Punto)

¿Cuál de los siguientes es un ejemplo de cómo el análisis de riesgos del producto puede influir en la minuciosidad y el alcance de las pruebas?

- a) El seguimiento continuo del riesgo nos permite identificar el riesgo emergente lo antes posible.
- b) La identificación de riesgos nos permite implementar actividades de mitigación de riesgos y reducir el nivel de riesgo.
- c) El nivel de riesgo evaluado nos ayuda a seleccionar el rigor de las pruebas.
- d) El análisis de riesgos nos permite derivar partidas de cobertura.

Seleccione una opción.

Solución

- a) *No es correcto. El monitoreo de riesgos es parte del control de riesgos, no del análisis de riesgos.*
- b) *No es correcto. La identificación de riesgos en sí misma no nos permite implementar actividades de mitigación de riesgos. Las acciones mitigadoras se definen durante la fase de control de riesgos.*
- c) *Es correcto. Este es un ejemplo de cómo el análisis de riesgos influye en la minuciosidad y el alcance de las pruebas.*
- d) *No es correcto. Los elementos de cobertura se obtienen mediante técnicas de prueba, no mediante análisis de riesgos.*

Mirar sección 6.2.3.

36. Pregunta 36 (1 Punto)

¿Cuál de las siguientes actividades en el proceso de prueba hace mayor uso de los informes de progreso de la prueba?

- a) Diseño de prueba.
- b) Finalización de la prueba.
- c) Análisis de pruebas.
- d) Planificación de pruebas.

Seleccione una opción.

Solución

- a) *No es correcto. Los informes de progreso de las pruebas se utilizan principalmente durante el monitoreo y control de las pruebas, y durante su finalización, no durante el diseño de las pruebas.*
- b) *Es correcto. Un informe de finalización de la prueba se prepara durante la finalización de la prueba, cuando un proyecto, nivel de prueba o tipo de prueba está completo y cuando, idealmente, se han cumplido sus criterios de salida. Este informe utiliza información de los informes de progreso de las pruebas y otros datos.*
- c) *No es correcto. Los informes de progreso de las pruebas se utilizan principalmente durante el seguimiento y control de las pruebas, y durante su finalización, no durante el análisis de las pruebas.*

- d) *No es correcto. Los informes de progreso de las pruebas se utilizan más durante el monitoreo y control de las pruebas, y durante su finalización, no durante la planificación de las pruebas.*

Mirar sección 6.3.2.

37. Pregunta 37 (1 Punto)

¿Cuál de los siguientes no es un ejemplo de cómo la gestión de la configuración respalda las pruebas?

- a) Todas las confirmaciones al repositorio están identificadas de forma única y controladas por versión.
- b) Se realiza un seguimiento de todos los cambios en los elementos del entorno de prueba.
- c) Todas las especificaciones de requisitos se mencionan sin ambigüedades en los planes de prueba.
- d) Todos los defectos identificados tienen un estado asignado.

Seleccione una opción.

Solución

- a) *No es correcto. Cuando un usuario informa de un fallo de software, gracias a la identificación única de los commits, es posible volver a ensamblar los archivos de la versión de software que utilizó el usuario (así como las versiones correspondientes de los scripts de prueba) y así reproducir el fallo, y localizar el defecto más rápido.*
- b) *No es correcto. Si un cambio en el entorno de prueba causa problemas inesperados durante las pruebas, la gestión de la configuración permite a los evaluadores retroceder a una versión anterior del entorno. Esto garantiza que las pruebas puedan continuar sin verse afectadas por el cambio.*
- c) *No es correcto. La gestión de la configuración garantiza que toda la documentación identificada (por ejemplo, especificaciones de requisitos) y elementos de software estén referenciados sin ambigüedades en la documentación de prueba (por ejemplo, planes de prueba).*
- d) *Es correcto. Esto lo garantiza la gestión de defectos, no el proceso de gestión de la configuración.*

Mirar sección 6.4.1.

38. Pregunta 38 (1 Punto)

Considere el siguiente informe de defectos para una aplicación de compras basada en web:

- Aplicación: WebShop v0.99
- Defecto: el botón de inicio de sesión no funciona.
- Pasos para reproducir:
 - Inicie el sitio web haga clic en el botón de inicio de sesión.
 - Resultado esperado: el usuario debe redirigirse a la página de inicio de sesión. Resultado real: el botón de inicio de sesión no responde cuando se hace clic.
 - Gravedad: Alta Prioridad: urgente

¿Cuál es la información más importante que falta en este informe?

- a) Nombre del evaluador y fecha del informe.
- b) Elementos del entorno de prueba y sus números de versión.
- c) Identificación del objeto de prueba.
- d) Impacto en los intereses de las partes interesadas.

Seleccione una opción.

Solución

- a) *No es correcto. Esto es importante, pero no tanto como los elementos del entorno de prueba.*
- b) *Es correcto. Lo importante que falta es la identificación del navegador y del dispositivo utilizado para la prueba. La información del navegador y del dispositivo es importante porque dicho defecto puede ser específico del navegador o del dispositivo. Por ejemplo, un botón de inicio de sesión puede funcionar bien en un navegador (o en una versión de un navegador específico) pero no en otro. Por lo tanto, la información del navegador y del dispositivo puede ayudar a los desarrolladores a reproducir el problema y encontrar la causa raíz del problema más rápidamente.*
- c) *No es correcto. Se identifica el objeto de prueba (WebShop v0.99).*
- d) *No es correcto. El impacto está incluido: esto es gravedad (alta).*

Mirar sección 6.5.1.

39. Pregunta 39 (1 Punto)

¿Herramientas de cuál de las siguientes categorías ayudan con la organización de casos de prueba, defectos detectados y gestión de configuración?

- a) Herramientas de ejecución y cobertura de pruebas.
- b) Herramientas de diseño e implementación de pruebas.
- c) Herramientas de gestión de defectos.
- d) Herramientas de gestión de pruebas.

Seleccione una opción.

Solución

- a) *No es correcto. Las herramientas de cobertura y ejecución de pruebas facilitan la ejecución automatizada de casos de prueba y la medición de la cobertura lograda al ejecutar esos casos de prueba. Sin embargo, estas herramientas no ayudan con la organización de defectos y la gestión de la configuración.*
- b) *No es correcto. Las herramientas de diseño e implementación de pruebas facilitan la generación de casos de prueba, datos de prueba y procedimientos de prueba, pero no ayudan con la organización de defectos y la gestión de la configuración.*
- c) *No es correcto. Las herramientas de gestión de defectos se utilizan para gestionar defectos, pero no son herramientas de prueba y no se utilizan para organizar casos de prueba o gestión de configuración.*
- d) *Es correcto. Las herramientas de gestión de pruebas aumentan la eficiencia del proceso de pruebas al facilitar la gestión del ciclo de vida de desarrollo de software (SDLC), los requisitos, las pruebas, los defectos y la gestión de la configuración.*

Mirar sección 7.1.1.

40. Pregunta 40 (1 Punto)

¿Cuál de las siguientes opciones es más probable que sea un beneficio de la automatización de pruebas?

- a) La capacidad de generar casos de prueba sin acceso a la base de prueba.
- b) El logro de una mayor cobertura mediante una evaluación más objetiva.

- c) El aumento de los tiempos de ejecución de pruebas disponibles con mayor potencia de procesamiento.
- d) La prevención de errores humanos mediante una mayor coherencia y repetibilidad.

Seleccione una opción.

Solución

- a) *No es correcto. Las herramientas de cobertura y ejecución de pruebas facilitan la ejecución automatizada de casos de prueba y la medición de la cobertura lograda al ejecutar esos casos de prueba. Sin embargo, estas herramientas no ayudan con la organización de defectos y la gestión de la configuración.*
- b) *No es correcto. Las herramientas de diseño e implementación de pruebas facilitan la generación de casos de prueba, datos de prueba y procedimientos de prueba, pero no ayudan con la organización de defectos y la gestión de la configuración.*
- c) *No es correcto. Las herramientas de gestión de defectos se utilizan para gestionar defectos, pero no son herramientas de prueba y no se utilizan para organizar casos de prueba o gestión de configuración.*
- d) *Es correcto. Las herramientas de gestión de pruebas aumentan la eficiencia del proceso de pruebas al facilitar la gestión del ciclo de vida de desarrollo de software (SDLC), los requisitos, las pruebas, los defectos y la gestión de la configuración.*

Mirar sección 7.2.1.