

Instrucciones:

- Cada pregunta múltiple opción contestada correctamente tiene un valor de 3,34 puntos. Cada pregunta incorrecta de la múltiple opción resta 1,11 puntos.
- **El puntaje total del examen es 100 puntos y se aprueba con 60 o más puntos.**

## Múltiple Opción

1. La empresa donde trabaja fue designada a desarrollar una nueva aplicación. Para la construcción de esta aplicación se recomienda utilizar una tecnología que no ha sido usada por ningún integrante del equipo. Investigaciones preliminares por parte del equipo de desarrollo sobre esta tecnología, revelaron que la curva de aprendizaje para poder utilizarla de forma adecuada es muy grande.

Seleccione la opción correcta:

- a) La utilización de esta tecnología es un riesgo negativo de este proyecto.
  - b) **(a) y una estrategia podría ser evitar el riesgo, desestimando la utilización de la tecnología y reemplazándola por alguna otra conocida.**
  - c) (a) y una estrategia podría ser mitigar este riesgo, desestimando la utilización de la tecnología y reemplazándola por alguna otra conocida.
  - d) La utilización de esta tecnología no es un riesgo de este proyecto.
2. Se ha designado a un equipo de desarrollo la construcción de una aplicación con un alto peso algorítmico. Esta aplicación es similar en lo que refiere a funcionalidad, tamaño y complejidad a una aplicación desarrollada hace un par de años por el mismo equipo de desarrollo. Este proyecto tiene la restricción de utilizar una tecnología de punta para la cual no hay experiencias pasadas ni en la empresa, ni en el mercado. Relevando en la empresa nos enteramos que Juan, desarrollador de otro equipo de la empresa, tiene experiencia en la tecnología aunque no conoce el sistema similar desarrollado anteriormente. ¿Qué técnica de estimación utilizaría?

Seleccione la opción correcta:

- a) LOCs.
  - b) **Juicio de experto por parte de los integrantes del equipo de desarrollo y Juan.**
  - c) Puntos de función.
  - d) Nivelación de recursos.
3. Un equipo está trabajando en un proyecto de 25 semanas de duración y un costo de USD 30.000. En la semana 12, calculan el SPI y el valor obtenido es 1,2.

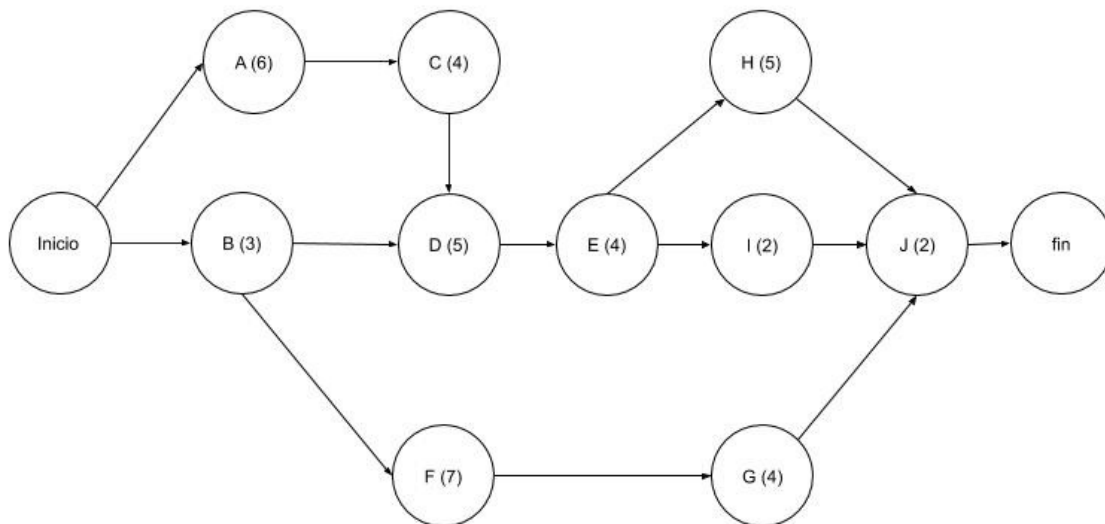
Seleccione la opción correcta:

- a) El proyecto está siendo más rápido de lo planificado.
  - b) **a) y si sigo con la misma tendencia el proyecto terminaría antes de la semana 21.**
  - c) El proyecto está costando menos de lo planificado.
  - d) c) y si sigo con la misma tendencia el proyecto va a costar menos de USD 27.000.
4. Dadas las siguientes afirmaciones:
- I. La EDT es la estructura donde se definen todas las actividades que van a ir en el cronograma, son las hojas del árbol de la EDT.
  - II. En el desarrollo ágil, el director de proyectos cumple el rol del *scrum master*.
  - III. Un proyecto sólo termina cuando el director de proyectos lo decide o si se termina el presupuesto.
  - IV. Una ventaja de la técnica *delphi* es que se pueden tener varias opiniones sin que hayan influencias entre los estimadores al no encontrarse en el mismo lugar físico.
  - V. La gestión de proyectos de distintas disciplinas por ejemplo civil y de software, no varía.

Seleccione la opción correcta:

- a) **Sólo una afirmación es correcta.**
  - b) Sólo dos afirmaciones son correctas.
  - c) Sólo tres afirmaciones son correctas.
  - d) Sólo cuatro afirmaciones son correctas.
5. La técnica de entrevistas individuales para la obtención de requisitos:
- a) Tiene como ventaja que es una técnica orientada a las personas y es poco flexible.
  - b) **Es efectiva para obtener información respecto a los requisitos y restricciones de la organización.**
  - c) a) y es muy útil en caso de tener cientos de usuarios que entrevistar para relevar los requisitos.
  - d) Todas las anteriores son correctas.

6. Sobre la ingeniería de software:
- Los métodos, herramientas y técnicas a utilizar en el desarrollo de software dependen únicamente del tipo de software a construir.
  - a) y un tipo de software son las aplicaciones independientes, que son aquellas aplicaciones que aunque funcionan en red, utilizan protocolos y tecnologías propietarias, y por tanto, independientes.
  - a) aunque los límites de los tipos de software muchas veces son borrosos y una aplicación puede corresponder a más de un tipo.
  - Ninguna de las anteriores es correcta.
7. Luego de presentar el cronograma de desarrollo, el cliente no está conforme con la fecha de entrega y solicita que el proyecto se termine 15 días hábiles antes, ya que quiere presentar el nuevo producto en un evento muy importante. Para evaluar la factibilidad de cumplir con la fecha solicitada, el equipo revisa los recursos y actividades a realizar y concluye lo siguiente:
- No es posible sumar un nuevo integrante al proyecto.
  - Algunos integrantes del equipo podrían realizar horas extras.
  - Revisando las tareas y el grafo de precedencias, presentado a continuación, el equipo resuelve que es viable que las tareas A y D se realicen de forma paralela.



Considerando las siguientes afirmaciones:

- Si el cliente acepta que parte del equipo realice horas extras, entonces se puede aplicar la técnica de Crashing.
- La cantidad de horas extras necesarias para poder terminar 15 días hábiles antes el proyecto es 120.
- Aplicar la técnica de Fast-Tracking puede generar un incremento en el costo del proyecto.
- La holgura del camino crítico va a ser 15 días menor.
- Las estrategias de Crashing y Fast-Tracking incrementan los riesgos del proyecto.

Seleccione la opción correcta:

- Sólo una opción es correcta.
  - Sólo (II) y (V).
  - Sólo (I), (III) y (V).
  - Sólo (III) y (IV).
8. La empresa en la que ud. trabaja está evaluando el desarrollo de una aplicación muy prometedora, pero que presenta una restricción importante en el presupuesto asignado.
- Seleccione la opción correcta:
- Una estrategia posible es realizar un estudio de factibilidad.
  - (a) y para realizar el estudio se necesita un conjunto de requisitos detallados por el equipo de desarrollo, para evaluar si es posible desarrollar la aplicación con los recursos disponibles y el presupuesto asignado.
  - (b) y como resultado del estudio se debe generar un informe que recomiende si vale la pena o no realizar el proceso de desarrollo del sistema.
  - Ninguna opción es correcta.

9. Respecto a los requisitos:
- Existen diferentes mecanismos para validar los requisitos, ejemplo de ellos son: entrevistas con el cliente, lectura por parte del cliente, y listas de comprobación.
  - a) y otra forma es elaborar prototipos.**
  - a), sin embargo, los prototipos no se utilizan para validar requisitos, su función es únicamente facilitar el diseño de los casos de prueba.
  - Los requisitos no funcionales son más fáciles de verificar que los requisitos funcionales ya que en general son transversales a todo el sistema y no dependen de una funcionalidad en particular.
10. Respecto a los modelos estructurales en ingeniería de requisitos:
- Son muy poco útiles como documentación del sistema.
  - Sirven como punto de partida para la definición de la arquitectura del sistema.
  - Deben especificarse con un alto nivel de detalle para evitar errores en la implementación.
  - Se debe utilizar el estándar de notación UML.
  - Al inicio de la fase de requisitos son muy útiles para representar entidades del negocio a nivel conceptual.

Seleccione la opción correcta:

- Todas las opciones son correctas.
  - Solo (II) y (V) son correctas.**
  - Solo (I), (II) y (V) son correctas.
  - Solo (I), (II), (IV) y (V) son correctas.
11. Respecto a la gestión y especificación de requisitos:
- El nivel de detalle con que se especifican los requisitos no difiere con respecto a los involucrados que los leen/usan, sino que depende de otros factores como el tamaño del sistema y su nivel de complejidad.
  - Los requisitos no funcionales son restricciones y propiedades del sistema que generalmente se establecen para cada funcionalidad particular que éste provee.
  - En el marco de las metodologías ágiles, los cambios en requisitos no funcionales se incorporan fácilmente y con poco impacto en el desarrollo del sistema.
  - Los casos de uso constituyen una técnica adecuada y ampliamente utilizada para modelar tanto requisitos funcionales como no funcionales.
  - Si se elige la técnica de casos de uso para la especificación de requisitos, en la descripción de los flujos deben reflejarse todos los detalles de la interacción con el sistema, incluyendo aspectos de la interfaz gráfica.

Seleccione la opción correcta:

- Solo (I), (II) son correctas.
  - Solo (I), (II), (III) son correctas.
  - Solo (IV) y (V) son correctas.
  - Ninguna de las opciones es correcta.**
12. En lo que respecta al Diseño
- Dado que es una actividad de base creativa, no puede reducirse a una serie de pasos a seguir.
  - a) Un objetivo a perseguir es lograr manejar la complejidad, ya que se reducen los costos del diseño.**
  - b) y aunque se logre manejar la complejidad no se reduce la posibilidad de introducir defectos durante el diseño.
  - b) Un camino para la reducción de la complejidad es el basarse en "estándares abiertos", más conocidos como "el principio abierto-cerrado".
13. Para crear la arquitectura de software se deben considerar los requisitos del sistema a construir. ¿Cuál de las siguientes afirmaciones es válida?
- No se consideran los requisitos funcionales.
  - No se consideran los requisitos no funcionales.
  - Se consideran tanto los requisitos funcionales como los no funcionales, basándose fuertemente en los funcionales.
  - Se consideran tanto los requisitos funcionales como los no funcionales, basándose fuertemente en los no funcionales.**
14. Ud. ha sido designado Arquitecto de un proyecto de desarrollo de software de gran magnitud:
- Manejará principios de diseño como: "principio abierto-cerrado", "alta cohesión" y "bajo acoplamiento".**
  - a). Será necesario que defina, en conjunto con el Jefe de Proyecto, los procedimientos de gestión de la configuración para los artefactos que genere.
  - Le indicará al cliente qué requisitos no funcionales debe cumplir la solución, ya que estos son críticos para la definición de la arquitectura.
  - Dos opciones son correctas

15. Sobre el modelado del sistema:
- No es necesario que los modelos cubran todo el sistema para cumplir con que los requisitos sean completos.
  - Los modelos sirven tanto para elaborar y explorar requisitos así como para diseñar soluciones de software.
  - a) y b) son correctas.**
  - Todas las anteriores son incorrectas.
16. Sobre la reutilización. ¿Cuál de las siguientes afirmaciones es correcta?
- Cierta cantidad del trabajo de diseño se realiza en construcción. A su vez, tanto las pruebas unitarias como parte de las pruebas de integración se realizan durante la construcción.
  - La anticipación al cambio afecta como el software es construido. En este sentido, un ejemplo es guardar los estados posibles de las facturas en un tabla de la base de datos o en un archivo de configuración en lugar de ponerlos fijos en el código.
  - La reutilización, por lo general, permite reducir los costos de la documentación aunque no del desarrollo.
  - Dos de las anteriores son correctas.**
17. Suponga un sistema de gran porte que se encuentra en producción. Actualmente existe un conjunto de cambios en los requerimientos para gran parte de sus funcionalidades, los cuales ya fueron analizados y se decidió implementarlos. Con respecto a la estrategia para su implementación:
- No tiene sentido llevar adelante un análisis de impacto ya que quien se encargue de la implementación de los cambios deberá entender todo el negocio y por lo tanto examinar todo el sistema.
  - El análisis de impacto es una actividad importante ya que facilita la estimación del esfuerzo de implementación.
  - El análisis de impacto cumple una función de relevancia en la evolución de este tipo de sistemas, siempre y cuando la implementación de los cambios esté a cargo del mismo equipo que desarrolló el sistema.
  - Para que la implementación de los cambios sea exitosa es necesario que al iniciar la implementación, los requerimientos estén especificados en forma detallada para no invertir tiempo en su refinamiento.
  - Antes de comenzar a implementar los cambios es importante priorizarlos y planificar las diferentes versiones de acuerdo a las prioridades asignadas.

Seleccione la opción correcta:

- Solo (I) y (V) son correctas.
  - Solo (II), (IV) y (V) son correctas.
  - Solo (II) y (V) son correctas.**
  - Solo (I), (IV) y (V) son correctas.
18. En el contexto de la pregunta anterior, suponga que una de las funcionalidades del sistema interactúa con un sistema heredado de la misma organización. Con respecto a la evolución de este sistema, seleccione la opción correcta:
- Cualquiera sea el sistema heredado, la mejor estrategia es reemplazarlo por una aplicación basada en nuevas tecnologías.
  - La estrategia a seguir depende de la calidad y del valor del sistema heredado para el negocio.
  - b) y en caso que la calidad sea muy buena y represente un alto valor para el negocio, una buena opción es continuar manteniendo el sistema.**
  - Para tomar la decisión de mantener el sistema heredado, el factor determinante es contar con técnicos que posean conocimientos en la tecnología base del sistema.

19. En cuanto a la liberación de un sistema. Seleccione la opción correcta:
- Es recomendable pensar y validar la estrategia de liberación desde el comienzo del proyecto.
  - Luego de liberado el sistema todos los recursos humanos involucrados en su desarrollo quedan disponibles para reasignarse a otros proyectos.
  - Se debe tener en cuenta los distintos tipos de usuarios que el sistema pueda tener, para entrenarlos y explicarles cómo utilizar el sistema en caso de ser necesario.
  - Dos de las anteriores son correctas.**
20. Con respecto a los modelos de procesos:
- En los procesos ágiles la planificación es incremental y se busca que sea flexible ante el cambio para poder reflejar los cambios en los requisitos y necesidades del cliente.
  - (a) y el desarrollo incremental puede aplicarse tanto a procesos ágiles como a procesos dirigidos por planes.**
  - Una de las ventajas del desarrollo incremental es que facilita el mantenimiento de la trazabilidad de documentos entre una versión y otra.
  - (c) y una desventaja del modelo de integración y configuración es que se pierde el control sobre la evolución de los elementos que se reutilizan del sistema.

21. La empresa para la cual trabaja es contratada para hacerse cargo de un proyecto de desarrollo de software que ya está en curso. Lo primero que se le pide como líder de SCM es que diseñe una estrategia de manejo de versiones para el desarrollo de una aplicación móvil. Se estima que se tendrá una liberación menor (solamente con bug-fixes) cada dos semanas y una mayor (con nuevas funcionalidades) cada dos meses. A su vez, se debe poder agregar y corregir funcionalidades a distintas versiones de forma independiente. Sin embargo, las correcciones de errores deben propagarse a todas las versiones existentes.

Indique la afirmación correcta.

- Usted elige Github como repositorio y define un esquema de versionado donde se tiene una única rama donde trabajan todos los desarrolladores en el código correspondiente a cada versión.
  - Introduce el concepto de branching, el cual implica la creación de una nueva versión de un componente de software al combinar versiones separadas en diferentes líneas de código.
  - Como se tienen cambios tan seguido y se está llevando a cabo una metodología ágil no es necesario disponer de ítems de configuración ni llevar documentado el contenido de cada versión, solo importa la que se está liberando.
  - Ninguna es correcta.**
22. Los cambios son inevitables en los grandes proyectos de software. En relación a esto, considere las siguientes afirmaciones:
- El costo de un cambio incluye el costo del retrabajo asociado a ese cambio (como por ejemplo volver a analizar los requisitos), así como el costo de la implementación de esa nueva funcionalidad.
  - Desarrollar un prototipo del sistema que muestre algunas funcionalidades claves a los clientes para su validación es un ejemplo de estrategia para anticiparse al cambio.
  - Una alternativa para reducir los costos del retrabajo es tener un proceso que tenga buena tolerancia ante los cambios. O sea, un proceso de desarrollo que permita que los cambios puedan ser llevados a cabo con un costo relativamente bajo.
  - Los prototipos son siempre desechables. Por tal motivo no es necesario documentar los mismos.
  - Si se utiliza el enfoque de entregas incrementales entonces las entregas tempranas cumplen la función de prototipo que ayudan a validar y relevar requisitos para próximos incrementos.

Seleccione la opción correcta:

- Todas son correctas.
  - Sólo (I), (II) y (V) son correctas.
  - Sólo (I), (II), (III) y (V) son correctas.**
  - Sólo (II), (III) y (IV) son correctas.
23. En cuanto a los métodos ágiles:
- Uno de los principios de los métodos ágiles trata sobre el involucramiento del cliente y busca que los clientes estén estrechamente involucrados a lo largo del proceso de desarrollo, priorizando los requisitos y validando los resultados de las iteraciones.
  - Uno de los principios del desarrollo ágil hace énfasis en la importancia del proceso y de las herramientas.
  - En el uso de Scrum es muy importante que el equipo se autogestione.
  - Extreme Programming* (XP) es fácil de incorporar a las prácticas de la mayoría de las organizaciones.
  - En Scrum, en la reunión diaria (*daily meeting*) cada miembro del equipo describe su progreso desde la última reunión así como problemas que hayan aparecido y lo que planifica para su jornada.

Seleccione la opción correcta:

- Todas son correctas.
  - Sólo (I), (III) y (V) son correctas.**
  - Sólo (II), (III), (IV) y (V) son correctas.
  - Sólo (I), (III) y (IV) son correctas.
24. Sobre los procesos de software:
- Cuando se describe un proceso se suelen describir las actividades principales del mismo y el orden en el cual se realizan las actividades.**
  - (a) y las descripciones de los procesos no deben incluir roles o responsabilidades de las personas que participan en el mismo, ya que eso depende de cada metodología.
  - Si se tiene los requisitos correctamente especificados y validados por el cliente, no es necesario luego contar con un proceso de desarrollo que considere la gestión de los cambios.
  - (c) y los prototipos se pueden utilizar para demostrar conceptos, probar opciones de diseño así como para conocer más sobre un problema y las posibles soluciones.

25. Sobre la mejora de procesos:
- Es necesario conocer el proceso con el cual se construye el software actualmente para poder mejorarlo.
  - Una de las claves para poder realizar una mejora es medir diversos aspectos del proceso actual para entender qué es necesario mejorar (y qué no). Para esto, no importa si el proceso está definido o no, ya que no influye en el proceso de medición.
  - Las medidas tomadas deben recolectar datos cualitativos en lo posible.
  - Luego de realizar un cambio en el proceso es necesario volver a medir el proceso para evaluar si la introducción de dicho cambio fue beneficiosa o no. Y en caso de que no, revertirlo o ajustarlo.
  - Una métrica muy común utilizada es la cantidad de defectos, que puede indicar: calidad del producto de software, calidad del proceso de verificación, calidad del proceso de construcción del software, entre otros.

Seleccione la opción correcta:

- Sólo (I), (II) y (III) son correctas.
  - Sólo (I) y (IV) son correctas.
  - Sólo (I), (III) y (V) son correctas.
  - Sólo (I), (IV) y (V) son correctas.
26. Se le contrata como líder de verificación y validación (también llamado líder de testing o QA) para una organización y se incorpora de forma inmediata al proyecto. El proyecto está en sus fases iniciales, relevando y especificando los requisitos del sistema. Dentro de las primeras medidas que toma como líder, resuelve diseñar una estrategia de pruebas junto con algunos líderes de desarrollo. El software a construir se compone de un gran núcleo monolítico que contiene la mayor parte de las funcionalidades.
- Dado el contexto presentado, usted propone, de forma acertada, comenzar planificando las pruebas de rendimiento. Uno de los objetivos de estas pruebas es sobre-exigir el sistema y comprobar que el mismo no fallará nunca sin importar la carga a la que es sometido.
  - (a) y luego de realizar las pruebas de rendimiento es posible afirmar que el sistema falla poco en un intervalo de tiempo conocido.
  - Una posible estrategia es colocar el foco inicial en las pruebas de las funcionalidades asociadas al bloque monolítico. Como cuentan con la especificación de historias de usuario, propone un diseño de casos de prueba temprano a partir de la especificación de dichas historias.
  - c) y en términos de pruebas unitarias (o de componentes) propone el uso de TDD y logra su aceptación por el equipo de desarrollo. En TDD las pruebas se escriben y se automatizan antes de que se implemente el componente correspondiente. Finalmente, un beneficio claro es que las pruebas automatizadas pueden ser utilizadas como pruebas de regresión.
27. En el contexto de la pregunta anterior y continuando con su rol de líder de verificación y validación, usted decide introducir la técnica de inspecciones dentro del ciclo de desarrollo del software, entendiendo que se dan las características de contexto apropiadas para introducirla.
- Para esto se construyen *checklists* que indiquen criterios de revisión sobre diferentes aspectos y den soporte al proceso. Cada revisor se debe centrar en los distintos aspectos específicos indicados en la checklist. El objetivo es encontrar defectos y elementos que no estén claros. El proceso permite medir la calidad del objeto bajo revisión y mejorar la calidad del propio proceso.
  - a) y esto lo puede hacer de forma unilateral sin lograr el consenso dentro de los líderes de desarrollo ya que es usted es quien decide en relación a verificación y validación. La efectividad y adopción de la técnica no requiere del involucramiento de las personas sino que basta con la adecuación al proceso.
  - Para que el costo de la inspección no impacte en el ciclo de desarrollo de forma negativa, se evitan las reuniones presenciales y el intercambio de comentarios se realiza a través de una herramienta.
  - a) y esto implica que durante la planificación del proyecto, hay que considerar la disponibilidad de integrantes calificados técnicamente que pertenezcan a equipos externos para que puedan realizar la inspección de forma estricta y sin sesgos. Finalmente, en la reunión de la inspección solo se discuten algunos puntos seleccionados y elegidos por un moderador.
28. Acerca de las estrategias de integración:
- Big-Bang es considerada una estrategia no incremental, mientras que Bottom-Up y Top-Down son consideradas estrategias incrementales.
  - a) y en Big-Bang se prueba cada módulo de forma aislada y luego se prueba la combinación de todos los módulos a la vez.
  - b) y en Bottom-Up se comienza por los módulos que dependen de otro(s) para ejecutar y se sigue hacia arriba según la jerarquía "usa".
  - a) y en Top-Down se requiere la construcción de "drivers" para poder realizarla.

29. De las técnicas de caja negra podemos afirmar que:
- I. Algunas de las técnicas que se pueden utilizar son particiones en clases de equivalencia, análisis de valores límites, pruebas basadas en casos de uso y pruebas basadas en la especificación.
  - II. El proceso de partición en clases de equivalencia incluye: identificar el dominio de entrada y salida, definir los casos de prueba para las clases de equivalencia del dominio de salida y generar la tabla de decisión para elegir el orden de ejecución.
  - III. No utilizan información interna de los componentes de software, sino que consideran el comportamiento del software desde el punto de vista de un observador externo y se basan en la especificación.
  - IV. Es difícil combinarlas con las técnicas de caja blanca, ya que ambas tienen perspectivas muy diferentes.

Seleccione la opción correcta:

- a) Sólo (I) es correcta.
  - b) Sólo (II), (III) y (IV) son correctas.
  - c) **Sólo (I) y (III) son correctas.**
  - d) Ninguna es correcta.
30. Las técnicas de pruebas basadas en la experiencia:
- a) Usualmente son técnicas de caja negra y siguen un enfoque sistemático.
  - b) Usualmente son técnicas de caja blanca y siguen un enfoque no sistemático.
  - c) Dependen de la experiencia del tester y son fuertemente influenciadas por la especificación de los requisitos
  - d) **Ninguna de las anteriores es correcta.**