

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. Según Ghezzi, la comprensibilidad es un atributo de calidad
  - a) Es una cualidad interna al producto en la medida que hace referencia a cómo un sistema está diseñado y escrito.
  - b) (a) y ayuda a lograr otros atributos de calidad como evolucionabilidad y verificabilidad.
  - c) Es una cualidad externa del producto en la medida que el producto está bien modularizado
  - d) (c) y es un componente de la amigabilidad del usuario.
2. Respecto al modelo de proceso Scrum:
  - a) Cuando los requisitos no son claros, es una buena oportunidad para aplicarlo.
  - b) En este modelo de proceso se fijan los recursos y el alcance; el tiempo queda variable.
  - c) (b), motivo por el cual es recomendable aplicar la técnica de *time-boxing*.
  - d) c), y para aplicar Scrum es necesario generar confianza con el cliente y que esté involucrado en el proyecto.
3. Comparado con el modelo de proceso en cascada, el desarrollo incremental tiene estos beneficios:
  - a) Se reduce el costo de adaptar los requisitos cambiantes del cliente.
  - b) (a) y el esfuerzo de reelaborar el análisis y la documentación ante un cambio suele ser menor que con el modelo en cascada.
  - c) Es más difícil obtener retroalimentación del cliente sobre el trabajo de desarrollo que se realizó.
  - d) Los clientes siempre tendrán la posibilidad de usar el producto final más temprano que con un modelo en cascada
4. Se desea desarrollar una aplicación web y móvil referente a la reserva de entradas a las mejores atracciones turísticas en todo el mundo. El cliente tiene altas expectativas del alcance de la aplicación, por lo que esta debe soportar grandes cargas de usuarios de forma concurrente.

Seleccione la opción correcta:

  - a) Un riesgo negativo es que el sistema no pueda soportar grandes cantidades de usuarios de forma concurrente
  - b) a) y para mitigar este riesgo es necesario contemplar este requisito al momento de la definición de la arquitectura.
  - c) Que el sistema soporte grandes cantidades de usuarios de forma concurrente es un requisito no funcional y, por lo tanto, no debe considerarse como un riesgo.
  - d) Ninguna opción es correcta.
5. Con respecto al WBS (estructura de desglose de trabajo):
  - a) Una de las ventajas de construir el WBS para un proyecto es que ayuda a gestionar el alcance del mismo.
  - b) (a) y sirve como base para las estimaciones y el armado del cronograma.
  - c) (b), y se construye de modo que cada nivel de descomposición contiene el 100 % del trabajo del padre.
  - d) (c), y una vez que el plan del proyecto esté en línea base, cualquier cambio en el WBS (que no sea un refinamiento) debe pasar por el control de cambios del proyecto.
6. Ya han transcurrido 3 meses desde el inicio del proyecto y el gerente de proyecto ha calculado los siguientes indicadores de valor ganado para saber la situación actual: SPI = 1,10 y CPI = 0,91.
  - I. El proyecto está saliendo más caro de lo previsto.
  - II. El proyecto se encuentra adelantado respecto a la planificación.
  - III. No se puede concluir nada sobre lo que se lleva gastado en el proyecto.
  - IV. La sobreestimación de tareas puede ser una de las causas para encontrarnos en la situación actual.
  - V. El proyecto se encuentra atrasado.

Seleccione la opción correcta:

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

7. La planificación se realiza durante las siguientes etapas en un ciclo de vida del proyecto:
- I. Durante la fase de inicio, cuando se debe determinar quién trabajará en el proyecto y cómo se asignarán los recursos.
  - II. En la etapa de propuestas; por ejemplo, cuando se decide presentarse a una licitación, con vistas a obtener un contrato para desarrollar un sistema de software, es de gran utilidad para decidir si se cuenta con los recursos para el trabajo.
  - III. Periódicamente a lo largo del proyecto, en instancias de planificación indicadas en el cronograma del proyecto.
  - IV. Cuando sea necesario ajustar el plan a la luz de la experiencia obtenida.
  - V. Cuando sea necesario replanificar a causa de un incidente
  - VI. Cuando sea necesario replanificar a causa de un desvío significativo en la planificación original.

Seleccione la opción correcta:

- a) I y III son incorrectas.
  - b) II es correcta y IV es incorrecta.
  - c) Todas son correctas menos una.
  - d) Todas las afirmaciones (de la I a la VI) son correctas.
8. Ante un conflicto dentro del equipo de trabajo es recomendable:
- a) dejar que se resuelva solo con el correr del tiempo
  - b) dilatar abordarlo mientras la situación no se torne insostenible
  - c) abordarlo cuanto antes y generalmente en privado
  - d) comentárselo directamente al director como primera medida
9. Sobre casos de uso:
- I. El flujo principal muestra la secuencia de pasos más común que se lleva a cabo para la ejecución del caso de uso.
  - II. La relación de inclusión permite definir casos de uso que «ejecutan» otros casos de uso.
  - III. El sistema es un actor del caso de uso.
  - IV. Las postcondiciones describen el resultado de la ejecución del caso de uso.
  - V. Una postcondición también debe de incluir el estado del sistema en caso de falla.

Seleccione la opción correcta:

- a) Todas son correctas.
  - b) I, IV.
  - c) II, III, V.
  - d) I, II.
10. Con respecto a los requisitos no funcionales (RNF):
- I. Un único requisito no funcional, como por ejemplo un requisito de seguridad, puede generar una serie de requisitos funcionales relacionados. También puede generar requisitos que restrinjan otros requisitos ya existentes.
  - II. Los requisitos no funcionales pueden provenir de las características requeridas del software (requisitos del producto), de la organización que desarrolla el software (requisitos de la organización), o de fuentes externas
  - III. El siguiente RNF: «El sistema debe ser fácil de utilizar por el personal médico y debe organizarse de forma tal de minimizar los errores de usuario» está escrito de forma tal que es verificable.
  - IV. Siempre que sea posible, hay que escribir los RNF de forma cuantitativa. Esto permite que luego puedan ser probados de forma objetiva.
  - V. Los RNF no están relacionados con las propiedades emergentes del sistema.

Seleccione la opción correcta:

- a) Todas las opciones son correctas.
  - b) Sólo (I) y (II) son correctas.
  - c) Sólo (III) y (V) son correctas.
  - d) Sólo (I), (II) y (IV) son correctas.
11. Indique la opción más adecuada.
- a) El proceso de priorización de requisitos es muy importante. Permite entre otras cosas entender mejor las necesidades de los clientes y la importancia de cada requisito.
  - b) Para priorizar los requisitos se debe tener en cuenta las necesidades de los clientes, las relaciones de precedencias de los requisitos y el costo de satisfacer cada requisito.
  - c) Se debe evitar que el proceso de priorización termine con todos los requisitos con la misma prioridad, para cumplir con el objetivo de la priorización.

- d) Todas las anteriores son correctas.
12. En relación a los estándares de programación se tiene que tener las siguientes consideraciones generales
- es necesario tener definido un estilo de codificación
  - (a) y fuerte independencia del lenguaje de programación
  - (a) y el estándar de programación debe contribuir a la mantenibilidad del software.
  - (b) y priorizar el código legible.
13. Indique la opción más adecuada.
- Un caso de uso describe una secuencia de interacciones entre un sistema y un actor externo que resultan en un resultado de valor para el actor.
  - y son dependientes del método de diseño y del lenguaje que se utilice para la implementación.
  - y los casos de uso tienen que desarrollarse por completo de una vez.
  - Ninguna de las anteriores es correcta.
14. Con respecto a la mejora de procesos:
- La mejora de procesos implica comprender los procesos existentes y cambiar dichos procesos con el objetivo de aumentar la calidad del producto y/o reducir costos y tiempo de desarrollo.
  - Las mediciones del proceso son datos cualitativos sobre el proceso de software. Un ejemplo de esto es el número de personas/horas requeridos para construir un componente de software.
  - Existen dos tipos de métricas del producto: las estáticas y las dinámicas. Un ejemplo de métrica dinámica es el tamaño del código.
  - No existe un proceso de software ideal o estándar que aplique para todas las organizaciones o para todos los tipos de software. Por lo tanto, la mejora de procesos, no implica sólo adoptar métodos, herramientas o procesos genéricos. Siempre existen factores organizacionales específicos, procedimientos y normas que influyen en el proceso y que se deben considerar al momento de introducir propuestas de mejoras en los procesos utilizados.
- Seleccione la opción correcta:
- Todas las afirmaciones (de la I a la IV) son correctas.
  - Solo (I) y (II) son correctas.
  - Solo (II) y (IV) son correctas.
  - Solo (I) y (IV) son correctas.
15. Sobre la calidad del producto y la mejora de procesos:
- La calidad del proceso es un factor fundamental para la calidad del producto.
  - El ciclo de mejora de un proceso consta de tres etapas: Cambio, Medición y Análisis.
  - Otros factores que afectan a la calidad del producto son: la calidad del personal, la tecnología de desarrollo, los costos y plazos.
  - Todas las anteriores son correctas.
16. Sobre métricas de software:
- Uno de sus objetivos es utilizarlas para evaluar la calidad de un software
  - a) y detectar áreas de mejora para poder tomar decisiones al respecto
  - a) y siempre se toman medidas directas y objetivas sobre los productos de software
  - Existe una relación inequívoca entre las métricas estáticas y la calidad del producto
17. En relación a una arquitectura cliente-servidor:
- Suele ser la arquitectura más conveniente para sistemas de procesamiento masivo (o «por lotes»)
  - Suele ser más eficiente que utilizar esquemas de «tubos y filtros»
  - No debe utilizarse para sistemas de información de misión crítica
  - Ninguna de las afirmaciones es correcta
18. 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.

- d) Se consideran tanto los requisitos funcionales como los no funcionales, basándose fuertemente en los no funcionales.
19. En el contexto de la arquitectura o diseño de un sistema de información:
- No siempre es conveniente aplicar la estrategia «dividir y conquistar».
  - Bajo un diseño modular, cambiar un módulo evita cambios en los módulos relacionados.
  - Una mayor abstracción del diseño implica mayor claridad y mantenibilidad del sistema.
  - Dos opciones son válidas.
20. Durante el diseño de la arquitectura de software:
- se deben evitar seguir los procesos de la organización, ya que el diseño es fuertemente una actividad creativa
  - se definen los subsistemas que componen el sistema
  - (b), las interfaces, y las reglas de interacción entre ellos
  - Todas las opciones son correctas
21. Dado los ítems de configuración A, B y C; para el ítem A existen las versiones 1.0, 1.1 y 1.2; para el B se tiene la versión 1.0 y para el C las versiones 1.0, 1.1, 1.2 y 1.3.
- Una línea base es un conjunto de versiones de ítems de configuración que han sido establecidos como parte de ella.
  - Ejemplos de ítems de configuración son: documento de requisitos, plan de verificación y validación y diagramas de casos de uso.
  - La única versión posible para la línea base es la 1.0 que contiene las siguientes versiones de los ítems de configuración: A 1.0, B1.0 y C1.0.
  - Una versión de la línea base puede contener las versiones de los siguientes ítems de configuración: A 1.1, B1.0 y C 1.3.
  - Los ítems de configuración de la línea base se establecen al inicio del proyecto y no se pueden agregar ni eliminar ítems durante el desarrollo del proyecto.
- Seleccione la opción correcta:
- Solo las afirmaciones I), II) y III) son correctas.
  - Solo las afirmaciones I), II), IV) y V) son correctas.
  - Solo las afirmaciones I), II), IV) son correctas.
  - Ninguna opción es correcta
22. La gestión de configuración es la gestión de un sistema de software en evolución. Abarca la gestión de cambios, la gestión de versiones, el armado del sistema y la gestión de liberaciones. Indique cuál de las siguientes afirmaciones es correcta.
- La gestión de cambios implica evaluar las propuestas de cambios de los clientes del sistema y otras partes interesadas y el seguimiento de las diferentes versiones de componentes de software a medida que se realizan cambios en ellas.
  - La gestión de liberaciones es el proceso de crear una versión completa y ejecutable del sistema mediante la compilación y el linkeo de sus componentes, librerías externas y archivos de configuración.
  - La integración continua es ampliamente utilizada en proceso de desarrollo clásicos, especialmente en sistemas grandes y complejos.
  - Ninguna es correcta.
23. Las pruebas de caja blanca:
- Se aplica solo para probar funcionalidad
  - Se aplica a unidades de programas como, por ejemplo, a las rutinas o a las operaciones asociadas a un objeto.
  - Se aplica para estudiar el comportamiento, con foco en las entradas y sus salidas relacionadas.
  - Todas las afirmaciones son correctas.
24. Una empresa contratada para desarrollar un sistema identificó los siguientes módulos: A, B, C y D; el módulo B es usado por el módulo A y el módulo B usa los módulos C y D. Dada la complejidad del módulo B, la empresa decidió probarlo de forma aislada.
- Seleccione la opción correcta sobre las pruebas de integración del módulo B:
- Para realizar las pruebas de integración es necesario conocer las interfaces de los módulos C y D
  - a) y para probar el módulo B, se debe desarrollar un driver de A y dos stubs de C y D respectivamente.

- c) a) y para probar el módulo B, se debe desarrollar un stub de A y dos drivers de C y D respectivamente.  
d) Para realizar las pruebas de integración no es necesario considerar los módulos C y D.
25. Sobre las pruebas de regresión:
- a) Tienen como objetivo detectar fallas en el uso de recursos por parte del sistema
  - b) Buscan detectar faltas introducidas al realizar modificaciones al código,
  - c) (b) por lo que su aplicación es siempre al momento de realizar pruebas unitarias.
  - d) (c) y es conveniente tener estas pruebas automatizadas.
26. Acerca de la verificación de software:
- a) Las técnicas de verificación estática, en comparación con las dinámicas, han mostrado ser más efectivas en la detección temprana de defectos
  - b) (a) mientras que las pruebas basadas en las técnicas de verificación dinámica no requieren para su diseño del código construido.
  - c) Ejecutar en conjunto tanto técnicas estáticas como dinámicas en un mismo proyecto genera más costos que beneficios por lo que no es recomendable en proyectos de gran tamaño.
  - d) La verificación estática solo aplica a la verificación de código fuente.
27. En la generación de casos de prueba a partir de casos de uso:
- a) Se deben tener en cuenta los flujos alternativos del caso de uso
  - b) (a) y los flujos alternativos generales, (por ejemplo: «G1. El usuario cancela la operación») deben incluirse en un único caso de prueba.
  - c) (b) y es necesario al menos cubrir todas las sentencias del código.
  - d) Ninguna de las anteriores es correcta
28. Las leyes de Lehman, referentes a la evolución de los sistemas, dicen que:
- a) Un programa utilizado debe ser adaptado continuamente; si no, deja de ser útil.
  - b) a) y con los cambios va a tender a ser menos complejo.
  - c) b) y la calidad del sistema va a declinar salvo que se realicen mejoras para mejorar.
  - d) c) y la evolución del programa es un proceso de autoregulación.
29. En cuanto a la relación entre el esfuerzo de desarrollo de software y el esfuerzo de mantenimiento de software, podemos afirmar:
- a) El esfuerzo en mantenimiento de un sistema es directamente proporcional al esfuerzo en desarrollo de ese sistema.
  - b) a) y si realiza mantenimiento preventivo aplicando refactorización, el costo de mantenimiento irá disminuyendo con el tiempo.
  - c) a) y esto es cierto solo al comienzo de operación del sistema. Conforme pasan los meses el software envejece por lo que los costos crecen exponencialmente.
  - d) Dedicar un mayor esfuerzo durante la etapa de desarrollo podría ayudar a un menor costo del mantenimiento del sistema.
30. La predicción de mantenimiento se ocupa de evaluar qué partes del sistema pueden causar problemas y cuáles tienen altos costos de mantenimiento,
- a) y se puede dividir en predicción de mantenibilidad, predicción de costos de mantenimiento y predicción de cambios al sistema.
  - b) a) y la predicción de cambios muestra directamente qué partes del sistema serán las más costosas de mantener.
  - c) b) y las métricas de complejidad permiten evaluar la volatilidad de los componentes del sistema.
  - d) a) y las métricas de proceso pueden ser utilizadas para evaluar la mantenibilidad.