

- 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 evolución de software es relevante debido a que:
  - I. Es necesario que el software construido se adapte a las nuevas necesidades del usuario
  - II. Comprende, por lo general, la mayor parte del ciclo de vida de un producto de software
  - III. El software requiere continua adaptación, mantenimiento para seguir siendo útil y valioso para la organización
  - IV. Los sistemas de software en general no representan activos críticos del negocio de las organizaciones.
  - a) Solo tres opciones son correctas.
  - b) (III) y (IV) son correctas.
  - c) (II) es incorrecta.
  - d) (I) y (II) son incorrectas.
  
2. En el proceso de evolución de software se encuentran las siguientes actividades:
  - a) Análisis del impacto, planificación de la versión
  - b) (a) e implementación del cambio
  - c) (b) y liberación de nueva versión
  - d) Ninguna de las anteriores.
  
3. Una organización perteneciente al sector financiero debe tomar una decisión acerca de los sistemas heredados:
  - a) Lo más apropiado y menos costoso es realizar un inventario de las funcionalidades de cada sistema y llamar a un concurso de proveedores de forma de sustituir todos los sistemas cuanto antes .
  - b) La estrategia a elegir depende de la calidad de los sistemas y del valor de cada sistema para el negocio.
  - c) (b) y al realizar la evaluación del valor para el negocio se debe considerar a los usuarios finales, a los clientes, gerentes y administradores.
  - d) (c) y si un sistema tiene baja calidad pero alto valor para el negocio, debe ser reestructurado o remplazado.
  
4. Tres equipos trabajan en un mismo proyecto de forma distribuida y en horarios diferentes. El director del proyecto considera que una posible gestión de versiones es:
  - a) mediante un control de versiones distribuido con enfoque de manejo de versiones entre pares
  - b) (a) ya que el riesgo de pérdida de datos es menor que en un enfoque centralizado
  - c) (a) a través de una única copia canónica del código
  - d) Ninguna de las anteriores es correcta
  
5. En relación a la etapa de identificación de riesgos:
  - a) Es el proceso de determinar los riesgos que pueden afectar al proyecto y documentar sus características
  - b) Este proceso no tiene por qué ser iterativo si al inicio del proyecto se identifican todos los riesgos de mayor impacto.
  - c) (a) y una técnica utilizada en esta etapa consiste en entrevistar a los participantes experimentados del proyecto, a los interesados y a los expertos.
  - d) (a) y para cada riesgo identificado se evalúa la probabilidad de ocurrencia y el impacto.

6. Dadas las siguientes afirmaciones:

- I. La velocidad de un equipo SCRUM expresa la cantidad de *story points* completados por *sprint*.
- II. En la técnica de Planning Poker es importante establecer la escala a ser utilizada.
- III. Para la creación del *Release Plan* no es necesario conocer la velocidad del equipo.
- IV. El *Sprint Burndown chart* muestra el progreso dentro del proyecto para alcanzar el objetivo del mismo.
- V. En Scrum, una de las responsabilidades del equipo de desarrollo es asegurar que al finalizar el *sprint* se tenga un producto potencialmente entregable.

Seleccione la opción correcta:

- a) Dos afirmaciones son correctas.
- b) Sólo (I), (II) y (V) son correctas.
- c) Sólo (III), (IV) y (V) son correctas.
- d) Sólo (I), (III) y (IV) son correctas.

7. En relación a la estimación en un proyecto de software:

- I. Es importante tener en cuenta experiencias pasadas de la organización (datos históricos).
- II. No es aconsejable estimar esfuerzo, ya que resulta muy variable dependiendo de la experiencia del equipo y del tipo de producto a desarrollar.
- III. La estimación debe realizarse al inicio del proyecto y actualizarse a lo largo del mismo de forma periódica.
- IV. Un proyecto debe re-estimarse cuando un miembro clave del equipo abandona el mismo.
- V. Es aconsejable estimar tamaño del producto en líneas de código, ya que promueve la producción de código de buena calidad.

Seleccione la opción correcta:

- a) Todas las opciones son correctas.
- b) Sólo (I), (II), (III) y (IV) son correctas.
- c) Sólo (III), (IV) y (V) son correctas.
- d) Sólo (I), (III) y (IV) son correctas.

8. En relación a las etapas de formación de un equipo (Tuckman & Jensen, 1977) vistas en el curso:

- a) Las etapas son: formación, brainstorming, normalización, desempeño y disolución.
- b) (a) y la duración de cada una de las etapas puede depender del tamaño y liderazgo del equipo
- c) (b) y equipos que ya trabajan juntos pueden saltarse etapas
- d) Ninguna de las anteriores es correcta

9. ¿Qué factores influyen en la estimación de costos de un proyecto?

- I. El tamaño estimado del producto
- II. El tamaño del equipo de desarrollo
- III. La experiencia del equipo de desarrollo
- IV. Las restricciones tecnológicas con las que debe ser construido el producto
- V. El presupuesto con el que se cuenta

Seleccione la opción correcta:

- a) Sólo (I), (II) y (IV) son correctas.
- b) Sólo (I), (II), (III) y (IV) son correctas.
- c) Sólo (II), (III), (IV) y (V) son correctas.
- d) Todas las opciones son correctas.

10. En lo visto en el curso sobre los productos genéricos y los productos específicos de software:

- a) Los productos genéricos corresponden a sistemas autónomos que se comercializan y venden a cualquier cliente que desee comprarlos.
- b) Los productos específicos corresponden a software desarrollado alrededor de una funcionalidad específica.
- c) La distinción entre productos genéricos y productos específicos radica en quién controla la especificación.
- d) Dos de las anteriores son correctas.

11. Sobre el modelado del sistema.
- Se recomienda que los modelos cubran todo el sistema para chequear 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.
12. Respecto a las técnicas para obtener requisitos:
- Uno de los usos de las entrevistas es entender el problema de negocio.
  - Las entrevistas cerradas se basan en una lista de preguntas pre determinadas.
  - Cuando se usa la técnica de Focus Groups es necesario contar con el rol de facilitador y realizar una buena selección de los participantes.
  - Las historias de usuario solo son útiles para especificar requisitos y no para obtenerlos.
  - La técnica de prototipado solo es útil para definir la interfaz de usuario.
- Seleccione la opción correcta:
- Sólo (I), (II) y (V) son correctas.
  - Sólo (I), (II) y (III) son correctas.
  - Sólo (II), (III) y (IV) son correctas.
  - Sólo (I), (IV) y (V) son correctas.
13. Respecto a los criterios de aceptación de una historia de usuario:
- Ayudan a entender mejor el comportamiento del producto frente a diferentes situaciones,
  - (a) y sirven de guía para el desarrollo de los test de aceptación,
  - (b) y no es necesario definirlos de forma temprana.
  - (b) y disminuyen la incertidumbre del equipo.
14. Sobre los casos de uso indique la opción incorrecta.
- Los casos de uso no tienen por qué desarrollarse por completo de una vez.
  - Se recomienda incluir en los flujos detalles de la interfaz gráfica.
  - No se recomienda incluir información de diseño o desarrollo, ni tampoco sobre la definición de tipos de datos.
  - No es posible modelar todos los requisitos utilizando esta técnica.
15. Sobre la ingeniería de requisitos
- Los stakeholders (o interesados) generalmente no participan del estudio de factibilidad.
  - Los prototipos toman un paso tentativo en el espacio de soluciones.
  - (a) y (b) son correctas.
  - Ninguna de las anteriores es correcta.
16. Durante el diseño de la arquitectura de software:
- Se define el proceso que se va a utilizar durante todo el desarrollo del software
  - Se definen los subsistemas que componen el sistema
  - (b), las interfaces, y las reglas de interacción entre ellos
  - (a) y (c) son correctas
17. Indique cual de las siguientes NO es una ventaja del sistema arquitectónico Tubos y Filtros
- Es intuitivo para aquellos casos en que se puede pensar la aplicación como secuencias de procesamientos de datos.
  - Es sencillo agregar nuevas transformaciones de forma que el sistema evolucione.
  - Es posible reutilizar los filtros
  - Está especialmente indicado para utilizarse en la construcción de Sistemas Interactivos.

18. Dadas las siguientes afirmaciones respecto al diseño de software indique la correcta:
- Dividir y conquistar, sugiere dividir el producto a construir en partes más pequeñas para facilitar el diseño.
  - Abstracción permite al diseñador considerar el sistema a más bajo nivel considerando los detalles de cada componente.
  - Modularidad es la característica de un sistema en el cual los componentes se pueden implementar separadamente y el cambio en uno tiene impactos mínimos en el resto.
  - Dos de las anteriores son correctas.
19. Según Sommerville, son ventajas de una arquitectura explícita:
- Mejorar la comunicación con los stakeholders.
  - Validar el cumplimiento de los requisitos no funcionales.
  - Favorecer la reutilización.
  - Todas las anteriores son correctas.
20. La construcción de software:
- Refiere a la creación detallada del software
  - (a) en la que se combina codificación, verificación y pruebas unitarias,
  - (b) así como pruebas de integración y debugging
  - Ninguna opción es correcta
21. Teniendo en cuenta los procesos de software:
- Cuando se describe un proceso se suelen describir las actividades principales del mismo y el orden en que dichas actividades se realizan.
  - (a) y las descripciones de los procesos pueden también incluir los roles o responsabilidades de las personas que participan en el mismo.
  - 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.
22. En relación a los modelos de procesos:
- Los modelos de procesos son una representación simplificada y de alto nivel de los procesos de software.
  - El RUP es un modelo de procesos universal por lo que se recomienda fuertemente su uso para todo tipo de software y para toda clase de equipos de desarrollo.
  - El modelo en cascada no resulta adecuado cuando se tiene comunicación informal entre el equipo y con los interesados (*stakeholders*).
  - Los métodos ágiles son métodos de desarrollo incremental.
  - Una de las ventajas del modelo de Integración y Configuración es que se reduce la cantidad de software a desarrollar, permitiendo por lo general entregar el software de forma más rápida.
- Seleccione la opción correcta:
- Todas las opciones son correctas.
  - Sólo 3 opciones son correctas.
  - Sólo (I), (II), (III) y (IV) son correctas.
  - Sólo (I), (III), (IV) y (V) son correctas.

23. Teniendo en cuenta las actividades básicas que se realizan en los procesos de software:
- En los métodos ágiles la especificación de requisitos no se ve como una actividad separada, sino que se ve como parte del desarrollo.
  - El diseño de la arquitectura y de la base de datos son actividades que suelen formar parte del proceso de diseño.
  - En los métodos ágiles no se realizan actividades de diseño, ya que se hace refactoring cuando es necesario y esto se hace directamente a la hora de codificar.
  - Las actividades que se realizan en los procesos de software se organizan diferente de acuerdo al modelo de proceso de desarrollo usado.
  - Cómo gestionar la evolución es una actividad importante que se debe definir a la hora de definir un proceso de software.
- Seleccione la opción correcta:
- Todas las opciones son correctas.
  - Sólo (I), (II), (IV) y (V) son correctas.
  - Sólo (II), (III) y (IV) son correctas.
  - Sólo (I), (III), (IV) y (V) son correctas.
24. En una organización que desarrolla software de diversos tipos para terceros (aplicaciones móviles, sistemas de seguridad, sistemas bancarios, entre otros):
- Se debe usar el mismo proceso de desarrollo de software para todos los proyectos, esto facilita el tener datos históricos comparables
  - Para los sistemas bancarios es conveniente usar cascada puro
  - Se debe definir un proceso de desarrollo diferente para cada proyecto, ya que el cliente, el producto y los usuarios varían según cada proyecto
  - Ninguna de las afirmaciones anteriores es correcta.
25. Hoy en día las presiones del mercado obligan a los equipos de desarrollo a liberar software con más frecuencia y en cortos períodos de tiempo. En relación a este fenómeno, se presentan las siguientes afirmaciones:
- El realizar liberaciones constantes favorece la retroalimentación del equipo con el feedback del uso de la aplicación en producción, lo cual es muy valioso
  - Las liberaciones constantes favorecen las solicitudes de cambio por parte de los usuarios/clientes, las cuales pueden traer como consecuencia actividades de refactoring de la aplicación
  - Como las versiones se colocan en producción de forma rápida, la gestión de cambios se torna una actividad menos necesaria
  - Tener un proceso iterativo en este tipo de escenarios es casi que obligatorio
- Seleccione la opción correcta:
- Todas las opciones son correctas.
  - Sólo (I), (II), y (IV) son correctas.
  - Sólo (II), (III) y (IV) son correctas.
  - Sólo (I), (II) y (III) son correctas.
26. En relación a los tipos de pruebas:
- Las pruebas de regresión son necesarias en la fase de mantenimiento y en el desarrollo cuando se utilizan procesos iterativos incrementales
  - (a) y es necesario realizar pruebas de regresión para verificar que los cambios o agregados realizados a la nueva versión no hayan degradado o introducido fallas en el funcionamiento del sistema.
  - Las técnicas de caja blanca son técnicas estáticas ya que analizan el producto para deducir su correcta operación.
  - (c) y el análisis estático generalmente es realizado por herramientas (por ejemplo: compiladores).

27. En relación a los niveles de pruebas:

- I. En las pruebas unitarias (también conocidas como pruebas de módulos o de componentes) se verifica el comportamiento de los componentes de acuerdo a su especificación.
- II. En las pruebas de integración se verifica si el sistema integrado (como un todo) cumple con los requisitos especificados.
- III. En las pruebas de sistema se verifica que el sistema cumpla con las necesidades y expectativas del cliente.
- IV. Para poder hacer las pruebas de sistema es necesario tener una visión global y conocer los requerimientos.
- V. Para realizar las pruebas de integración es necesario conocer las interfaces.

Seleccione la opción correcta:

- a) Sólo (I), (II) y (IV) son correctas.
- b) Sólo (I), (IV) y (V) son correctas.
- c) Sólo dos opciones son correctas.
- d) Sólo (I), (III), (IV) y (V) son correctas.

28. Elementos de un caso de prueba son:

- I. Objeto de prueba
- II. Entradas requeridas
- III. Salidas esperadas
- IV. Precondiciones (contexto en el cual debe ejecutarse)
- V. Pasos o procedimiento de ejecución
- VI. Artefactos de software a los cuales está relacionado (especificación de requisitos, diseño, arquitectura, etc)

Seleccione la opción correcta:

- a) Todas las opciones son correctas
- b) Sólo 5 opciones son correctas
- c) Sólo (II), (III), (IV) y (V) son correctas.
- d) Sólo (I), (II), (III) y (IV) son correctas.

29. Registrar y categorizar los defectos puede ser útil para:

- a) Mejorar la calidad del código
- b) (a) y mejorar la calidad del proceso de construcción del código
- c) (b) y mejorar el proceso de verificación
- d) (c) y mejorar la calidad del diseño

30. El orden en el cual realizar las pruebas depende de:

- I. El impacto que pueda tener en el cliente que la parte del software relacionada a esa prueba falle
- II. El costo de realizar esa prueba
- III. La dependencia de esa prueba con otras pruebas del sistema
- IV. La planificación de actividades del equipo de desarrollo
- V. La prioridad de esa parte del software definida por el cliente

Seleccione la opción correcta:

- a) Todas las opciones son correctas
- b) Sólo 4 opciones son correctas
- c) Sólo (I), (III) y (V) son correctas.
- d) Sólo (II), (III) y (V) son correctas.