

- Cada pregunta de la múltiple opción contestada correctamente tiene un valor de 3,34 puntos. Cada pregunta incorrecta de la múltiple opción resta 1,11 puntos.
- Para aprobar el examen se deben obtener al menos 60 puntos.

## Múltiple Opción

- 1) ¿Cuál de las siguientes NO es una ventaja de diseñar y documentar la arquitectura del software?
  - a. Favorece la comunicación entre los distintos stakeholders
  - b. Permite el reuso a gran escala.
  - c. Favorece la toma temprana de decisiones de diseño.
  - d. **Permite decidir tempranamente que tipo de proceso de desarrollo de software conviene utilizar para desarrollar el sistema.**

---

- 2) El estilo de arquitectura elegido para una aplicación depende fuertemente de
  - a. El proceso de software elegido para desarrollar la aplicación.
  - b. **Los requerimientos no funcionales del sistema.**
  - c. La cantidad de casos de uso identificados para el sistema.
  - d. Todas las anteriores.

---

- 3) El acoplamiento:
  - a. **Es un indicador sobre la dependencia entre módulos.**
  - b. (a) Además, el alto acoplamiento facilita la comprensión y mantenimiento de los módulos del sistema.
  - c. (a) y los procesos de desarrollo conocidos como “pesados” producen módulos con un correcto acoplamiento.
  - d. Es un indicador de qué tan cohesivo es un módulo

---

- 4) Respecto a la arquitectura de software
  - a. Una arquitectura mantenible generalmente resulta en una performance menor a la requerida
  - b. Una arquitectura con alta disponibilidad generalmente resulta poco segura.
  - c. Solamente se puede aplicar un único estilo arquitectónico a todo el diseño de la arquitectura de un sistema.
  - d. **Ninguna de las anteriores**

---

- 5) Acerca de las definiciones de error, defecto, falta y falla dadas en el curso
  - a. Falta y defecto tienen el mismo significado
  - b. Un error humano puede generar una falta y ésta a su vez puede generar una falla
  - c. Una falla es una manifestación externa de un defecto
  - d. **Todas las anteriores**

---

- 6) Respecto a las fallas en el software:

I) El software falla cuando no hace lo requerido o hace algo que no debería.  
II) Una razón por la que el software falla son las faltas en el diseño  
III) Una razón por la que el software falla son las faltas en el código

Son correctas las afirmaciones:

  - a. I
  - b. II y III
  - c. **I, II y III**
  - d. El software no falla

---

- 7) En el proceso de Verificación y Validación las técnicas estáticas de verificación
  - a. son efectivas en la detección temprana de defectos.
  - b. (a) y sirven para verificar no solo el código sino también los requerimientos y el diseño.
  - c. **(b) y la inspección de software es una de estas técnicas**
  - d. (c) y permiten tener en cuenta el ambiente donde será ejecutado el software.

---

- 8) Para diseñar un conjunto de pruebas de caja negra
  - a. se necesita el código que se va a verificar para que los casos puedan ejecutarse.
  - b. (a) y de esta forma asegurar que se ejecuten todas las sentencias
  - c. **se necesita la especificación de lo que se va a probar (objeto, programa, sistema).**
  - d. (c) y no se debe a comenzar el diseño de casos hasta no tener implementado el código

- 
- 9) ¿Cuál de las siguientes no es una estrategia de pruebas de integración?
- Big-bang
  - Por disponibilidad
  - Por complejidad mutua**
  - Top-down
- 
- 10) La Configuración del Software
- comprende ítems generados durante del proceso de ingeniería de software.
  - (a) y algunos ejemplos de ítems pueden ser: especificación de requerimientos, conjuntos de casos de prueba, reportes de defectos, manuales de usuarios.**
  - (b) y además esto quiere decir que desde que se crea un elemento de configuración es necesario controlar los cambios mediante procedimientos formales.
  - (b) y un elemento de configuración se convierte en línea base apenas se crea.
- 
- 11) Con respecto a la Gestión de la Configuración
- Los antipatrones mencionados en clase corresponden a situaciones que ocurren habitualmente en dicha disciplina y deben evitarse.
  - No es posible realizar la gestión de la configuración en un proyecto sin utilizar herramientas de software que apoyen las actividades involucradas.
  - (a) y algunas de las situaciones planteadas se pueden evitar utilizando herramientas para gestionar la configuración.**
  - Ninguna de las anteriores.
- 
- 12) Acerca de faltas y fallas
- Si un programa tiene una falta y se ejecuta durante un plazo suficientemente largo necesariamente se va a producir una falla.
  - Si luego de probar un programa durante un período suficientemente largo no aparece ninguna falla, esto significa que el programa carece de faltas.
  - Se puede considerar que el proceso de probar un programa consiste en tratar de que el programa falle y a partir de esa falla identificar faltas en el mismo.**
  - Si un programa que acepta dos datos de entrada y tiene que devolver como resultado uno de los valores “Si”, “No”, se prueba con 10000 parejas de datos de entrada distintos y falla con todas esas parejas, esto significa que el programa presenta al menos 10000 faltas.
- 
- 13) Entre los atributos de calidad relevantes para un juego para teléfonos celulares están:
- confiabilidad, seguridad de la información (security), seguridad de la integridad física (safety), facilidad de verificación.
  - adecuación al uso, amigabilidad de la interfaz de usuario, eficiencia en el uso de recursos, tiempo de respuesta.**
  - facilidad de aprendizaje, calidad de la documentación técnica, facilidad de mantenimiento, modularidad.
  - correctitud, robustez, confiabilidad, adecuación al uso.
- 
- 14) Respecto a modelos de proceso de software:
- El esquema ETVX solo es adecuado para especificar modelos de proceso prescriptivos.
  - La planificación de un proyecto de desarrollo de software es independiente del modelo de proceso que se esté aplicando, basta con tomar en cuenta la guía de PMBOK.
  - El contar con un modelo de proceso permite razonar sobre la forma mediante la cual construimos software, en particular para encarar la mejora del proceso, por ejemplo para generar productos de mejor calidad, a menor costo o en menor plazo.
  - (c) y una de las ventajas de un proceso en fases con liberaciones parciales en producción es que permite acortar el plazo para poder contar con resultados de utilidad, lo que a su vez permite atender la cada vez más buscada reducción del plazo para salir al mercado (time-to market).**
- 
- 15) El mantenimiento de software
- se puede clasificar en “correctivo”, “preventivo”, “perfectivo” y “adaptativo”.
  - (a) y el esfuerzo que dedica una organización a cada uno de estos distintos tipos de mantenimiento constituye un elemento de información relevante para evaluar la calidad del software bajo mantenimiento y también de la del propio proceso de mantenimiento.**
  - (b) y en general es deseable que una organización de mantenimiento dedique la mayor parte de su esfuerzo de mantenimiento al de tipo “correctivo” ya que esta es una vía para lograr obtener un software de mejor calidad a través de la corrección de los defectos que contiene el software.
  - (c), si bien en general existe cierta probabilidad de que se introduzca un defecto nuevo como resultado de una intervención de mantenimiento de tipo “correctivo”.
-

16) Dados los siguientes proyectos y modelos de proceso.

Proyectos:

A - Alto riesgo, gran tamaño (15.000 Puntos de Función), alcance ambicioso y con aspectos no del todo definidos, al menos dos subconjuntos de funcionalidad definidos que resultan útiles para los usuarios, involucra a múltiples organizaciones, algunas con dispersión geográfica y a más de 2000 usuarios, larga duración (un año y medio o más).

B – Riesgo medio, alcance definido, al menos tres subconjuntos de funcionalidad definidos que resultan útiles para los usuarios, duración media (ocho meses).

C – Riesgo bajo, simple, alcance definido, corta duración (cinco semanas).

Modelos de Proceso:

1 – Cascada.

2 – Especificación Operacional

3 – En Fases con Liberaciones Parciales en Producción y cada Fase con un Proceso Iterativo e Incremental.

4 – En Fases con Liberaciones Parciales en Producción y cada Fase de acuerdo al Modelo Transformacional.

Elija la asignación que le parezca más adecuada.

- a. **A3, B3, C1**
- b. A4, B2, C2
- c. A4, B2, C1
- d. A4, B3, C1

17) Respecto a la liberación del software

- a. Conviene comenzar su planificación cuando la arquitectura del producto está estable y es seguro que cumple con los requerimientos no funcionales.
- b. **En la planificación de la puesta en marcha de un producto conviene prever recursos destinados a la atención de problemas ya sea de software, hardware, instalación, problemas derivados de la falta de comprensión de cómo utilizar el producto por parte de los usuarios, y otros.**
- c. En general cuando un grupo de desarrollo construye un producto de software para un cliente, es el cliente quien se encarga de brindar el entrenamiento en el uso del producto a los distintos usuarios.
- d. (c) y si el producto o sistema nuevo sustituye a uno anterior que ya estaba en funcionamiento, en general es el propio cliente el responsable por la conversión de datos del sistema anterior al nuevo por lo que tanto el entrenamiento como la conversión de datos no suelen tener impacto en el alcance del proyecto.

18) Con respecto al Comité de Control de Cambios

- a. El objetivo del comité es controlar los cambios (por ejemplo mejoras o corrección de defectos).
- b. (a) y está integrado de forma tal que aúne las visiones de cliente, usuarios y desarrolladores.
- c. (b) y dentro de los pasos que realiza el comité se encuentran: calificar el cambio, evaluar la severidad y/o el impacto y su prioridad.
- d. **(c) y es conveniente que exista un procedimiento que asegure que quede registrado quién hizo el cambio, por qué se hizo y qué cambió.**

19) La capacidad de un proceso es:

- a. Una medida de la madurez de la organización.
- b. **La probabilidad de que los resultados obtenidos luego de su ejecución coincidan con los estimados.**
- c. Una medida de los resultados reales obtenidos luego de su ejecución.
- d. Ninguna de las anteriores.

20) Una diferencia conceptual entre CMMI e ISO/IEC 25000 es que:

- a. El primero es un estándar de procesos y el segundo una norma de productos.
- b. El primero se un modelo de evaluación de procesos y el segundo es un modelo de mejora de procesos.
- c. **El primero es un modelo de madurez de los procesos de una organización y el segundo es una guía para el uso de las normas SquaRE.**
- d. Es que CMMI es modelo de calidad de software e ISO/IEC 25000 un modelo de gestión de procesos de software.

21) En el contexto de la mejora y evaluación de procesos, se entiende por "proceso":

- a. **una secuencia de pasos ejecutados con un propósito.**
- b. (a), que deben ser definidas por el Encargado de SQA y validadas por el Director de Proyecto.
- c. (a). Estos procesos son utilizados por un programa en ejecución, y gestionados por el sistema que se esta construyendo en el Proyecto.
- d. (b). Los procesos pueden ser revisados y redefinidos con el objetivo de mejorar su calidad.

- 
- 22) ¿Cuál de las siguientes opciones puede ser considerada un ejemplo de proyecto?
- gestionar la operación continua de un servicio.
  - crear nuevos servicios.**
  - crear un producto y mantenerlo mientras esté en operación.
  - Todas las anteriores.
- 
- 23) La ejecución de un proyecto va a estar condicionada por
- la experiencia de los integrantes del equipo de desarrollo.
  - el conocimiento sobre las tecnologías que se deba utilizar en el desarrollo e incorporar en el producto
  - el proceso que se emplee en el desarrollo del proyecto.
  - Todas las anteriores.**
- 
- 24) Al intentar acortar la duración de un proyecto, sin reducir el alcance ni la calidad del producto, ocurre lo siguiente:
- es probable que se incremente el esfuerzo.**
  - contratar más personal reduce el esfuerzo de forma proporcional.
  - al paralelizar actividades se logra siempre reducir la duración del proyecto en un lapso igual al lapso en que se solapó la ejecución de las actividades paralelizadas.
  - la duración de un proyecto se puede reducir tanto como el administrador del proyecto quiera.
- 
- 25) ¿Cuál de las siguientes es una razón válida para cancelar un proyecto?
- Es claro que no se van a alcanzar los objetivos
  - La necesidad que lo motivó ya no existe
  - La ventana de mercado que lo motivó se cerró
  - Todas las anteriores**
- 
- 26) Si se utiliza la técnica de observación para relevar requisitos, se debe tener en cuenta el posible efecto Hawthorne, que consiste en:
- generalizar algo que es particular o local.
  - que la duración de las tareas se extiende tanto como los plazos permitan.
  - que las tareas se realizan siempre a último momento.
  - que el comportamiento de los empleados se modifica si se sienten observados.**
- 
- 27) Los "Casos de Uso":
- Son una técnica para recabar requisitos de los clientes y usuarios.**
  - (a). Al describirlos, deben considerarse tanto la totalidad de restricciones del proyecto, como el comportamiento esperado del sistema desde el punto de vista del usuario.
  - (b) y especifican qué es lo que hace el sistema, sin profundizar en cómo debe hacerlo
  - (c) y son secuencias de acciones en las que participan actores y el sistema, dando un resultado de valor observable para un actor particular.
- 
- 28) Entre las técnicas para relevar requisitos, la técnica de observación:
- se debe llevar adelante solo en períodos de funcionamiento normal.
  - consiste en realizar observaciones de forma espontánea, y se anota lo que surja en el momento.
  - se debe llevar adelante sin que los involucrados lo sepan.
  - tiene el problema de que los resultados no se pueden generalizar, porque pueden existir sesgos locales o particulares.**
- 
- 29) Indique cuál de los siguientes diagramas es un diagrama UML que se puede utilizar dentro de las técnicas de modelado de los requisitos y presenta una descripción estática del sistema:
- Diagramas de Actividad
  - Diagramas de Máquinas de Estado
  - Diagrama de Clases**
  - Todos los anteriores
- 
- 30) Dentro del proceso de administración de los requisitos, habiendo ya planificado el procedimiento de administración del cambio, la administración del cambio:
- comienza en paralelo al proceso de los requisitos.
  - comienza cuando se tiene una primera versión estable de los requisitos en línea base.**
  - implica que cualquier cambio solicitado por el cliente se debe implementar y se debe reflejar actualizando el documento de especificación de requisitos.
  - implica que se debe seguir el procedimiento de administración del cambio definido, solo si el cambio solicitado es de cierta magnitud.