

Instrucciones:

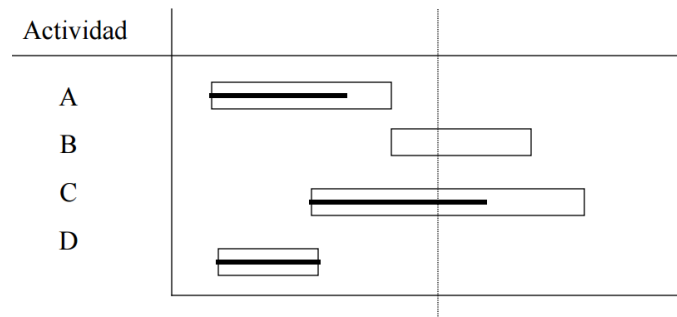
- Cada pregunta múltiple opción contestada correctamente tiene un valor de 3,34 puntos. Cada pregunta incorrecta de la múltiple opción resta 1,11 puntos.
- **El puntaje total del examen es 100 puntos y se aprueba con 60 o más puntos.**

Múltiple Opción

1. Sobre el mantenimiento perfecto:

- Busca mejorar algún aspecto ya presente
- Busca introducir cambios en respuesta a problemas que surgen en el uso diario
- (a) y según el estudio de Lientz-Swanson es el tipo de mantenimiento más común.**
- (b) y un problema puntual de este tipo de mantenimiento es la comprensión limitada del sistema.

2. Dado el siguiente diagrama Gantt con la planificación de actividades del proyecto y el avance real:



Si las tareas A y C están en el camino crítico:

- El proyecto está adelantado.
 - El proyecto está atrasado.**
 - El proyecto está en fecha.
 - La información dada no es suficiente para afirmar ninguna de las anteriores.
3. Se quiere desarrollar un sistema de software que controle la venta de entradas a diferentes eventos artísticos, realizados en Uruguay. El sistema debe permitir la compra de entradas por internet, a través de distintas redes de cobranzas (ejemplo: Abitab, redPagos, etc.) y en taquilla.

A continuación se detallan algunos puntos del documento de especificación de requisitos de software (SRS)

- Un usuario podrá buscar espectáculos.
- La compra por internet puede ser anulada hasta 3 horas antes del comienzo del espectáculo.
- Los altos volúmenes de información no deberán afectar la disponibilidad del sistema ni degradarán su performance.
- La probabilidad de falla del sistema no podrá ser mayor a 0,05. En caso de fallas o errores, el promedio de duración de los mismos no podrá ser mayor a 15 minutos.
- Se deberá validar de forma temprana la integración con la red de cobranza Abitab.

Seleccione la opción correcta:

- Los requisitos funcionales planteados en los puntos I) y II) son precisos.
- Los puntos III) y IV) describen requisitos no funcionales y ambos son verificables.
- El punto V) corresponde a una restricción del proyecto.**
- Todas las anteriores son incorrectas.

4. Cuando se libera un sistema que va a sustituir a otro en funcionamiento:
- se deben considerar distintas estrategias de conversión como: top-down, desatendida, big-bang y en paralelo
 - (a) y la carga de datos puede hacerse de forma manual o automática
 - (b) y al momento de ponerlo en funcionamiento se debe entregar toda la documentación de usuario y técnica.
 - Si se elige la estrategia de procesamiento en paralelo, se entiende que un sistema estará en producción y el otro en prueba y control.
5. El proceso de gestión del cambio:
- Es el proceso que se sigue en el análisis de requisitos de un cambio, e incluye las actividades que evalúan el impacto del cambio.
 - Debería aplicarse a todos los cambios propuestos a los requisitos, estén o no en la línea base.
 - Debería aplicarse solo a los cambios propuestos por el cliente.
 - Ninguna de las anteriores es correcta.
6. Con respecto a la evaluación y mejora de procesos:
- Un filtro de defectos es una actividad de aseguramiento de calidad que se introduce para eliminar una o más fuentes de defectos
 - (a) y en general tienen la forma de checklists (listas de verificación) que verifican si se siguió el procedimiento especificado en la realización de una o más actividades
 - (b) y se deben introducir filtros de defectos en todas las actividades definidas en el modelo de proceso.
 - Ninguna de las anteriores es correcta.
7. La mejora del proceso significa:
- comprender los procesos existentes
 - (a) y cambiarlos para mejorar la calidad del producto
 - (b) y aumentar el tiempo de desarrollo.
 - (b) y/o reducir los costos.
8. Con respecto a la Ingeniería de Software:
- Esta no aplica a procesos de desarrollo ágiles, dado que se centran principalmente en la construcción y no en la planificación del proyecto.
 - No justifica aplicarla para planificar proyectos si comenzamos desarrollando un prototipo evolutivo en el cual se va refinando según las necesidades del cliente.
 - Ayuda a definir el proceso de desarrollo, la planificación y estimar el esfuerzo total del proyecto, aunque este último suele variar durante el proyecto.
 - Toda construcción de software involucra realizar tareas de Ingeniería de Software.
9. 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.
10. El proceso de gestión de riesgos
- es un proceso que se aplica para proyectos con complejidad crítica
 - en un proceso que se aplica en la etapa inicial de proyectos cuyos requisitos están bien determinados
 - es un proceso que se aplica a lo largo de todo el proyecto
 - es un proceso que solo es necesario aplicar cuando los requisitos funcionales y no funcionales son inciertos

11. Sobre el costo del mantenimiento de software:

- I. Un equipo de mantenimiento estable puede reducir los costos del mismo
- II. Un software legado, construido con tecnología hoy obsoleta, puede aumentar los costos del mantenimiento, debido a la dificultad de contar con personal calificado en esa tecnología
- III. Con el paso del tiempo el software envejece y el mantenimiento disminuye, por lo cual también disminuye su costo
- IV. Sucesivas correcciones centradas en el corto plazo incrementan costos de mantenimiento subsiguientes
- V. Un equipo de desarrollo puede disminuir los costos del futuro mantenimiento, por ejemplo produciendo una documentación de buena calidad.

Seleccione la opción correcta:

- a) Solamente I, II, IV y V son verdaderas
- b) Solamente I, III y V son verdaderas
- c) Todas son verdaderas
- d) Solamente I y II son verdaderas

12. Acerca de la arquitectura y diseño puede decirse que:

- a) es una descripción de la organización de un sistema de software.
- b) (a) y algunas propiedades como la usabilidad de la interfaz de usuario, la disponibilidad, el rendimiento y la seguridad dependen de la arquitectura utilizada.
- c) (a) y algunos patrones comunes son: Modelo-vista-controlador, en capas, cliente-servidor, pipeline y pizarra.
- d) La única forma de documentar la arquitectura es mediante una vista de despliegue (vista física).

13. Frecuentemente, la mayor parte del trabajo en grupo se dedica a:

- a) trabajo individual
- b) trabajo no productivo
- c) interacción con otras personas
- d) resolución de conflictos

14. En cuanto a la Gestión de la Configuración (SCM) se puede decir que:

- a) Abarca gestión de cambios, gestión de las versiones, y gestión de las liberaciones.
- b) No es necesaria cuando se trabaja con equipos en el mismo lugar ya que la comunicación continua basta para conocer el estado de las versiones y decidir acerca de los cambios.
- c) También es necesaria luego de liberar un sistema, ya que una vez en uso se deberán atender los cambios que surjan.
- d) Dos de las anteriores son correctas.

15. Sobre los prototipos

- a) Los mock-ups son prototipos verticales, esto significa que implementan una porción funcional de la aplicación.
- b) (a) y permiten resolver incertidumbres sobre la factibilidad de la arquitectura propuesta u otros riesgos técnicos.
- c) Los mock-ups son prototipos horizontales, esto significa que se enfocan en porciones de la interfaz de usuario
- d) (c) aunque no permiten explorar comportamientos específicos del producto.

16. En una arquitectura orientada a servicios, un servicio web:

- a) es independiente de las aplicaciones que lo usan
- b) a) su descripción es independiente del lenguaje de programación utilizado
- c) b) y es recomendable utilizar dado que permite aumentar la cohesión de los sistemas
- d) Ninguna de las anteriores

17. Respecto a evaluación y mejora de productos y procesos:

- a) Un estándar provee un cuerpo de conocimiento que guía el desempeño de una disciplina profesional
- b) (a) y la capacidad de un proceso es la probabilidad de que los resultados obtenidos coincidan con las predicciones (estimaciones) de su desempeño
- c) (b) y en la mejora de procesos, el enfoque ágil no es adecuado para reducir el costo del proceso
- d) (c) y el paradigma GQM (Goal-Question-Metric) tiene como objetivo analizar el software y los procesos.

18. Dadas las siguientes afirmaciones respecto a las cualidades de software, según Ghezzi:

- a) La falta de la cualidad oportunidad, que tiene que ver con la adecuación al momento tecnológico, tuvo como consecuencia la llamada "crisis del software".
- b) La interoperabilidad refiere a la habilidad de un sistema de coexistir y cooperar con otros sistemas.
- c) La portabilidad consiste en que el sistema puede ser ejecutado en distintos ambientes, donde estos refieren a distintas plataformas de hardware, pero no incluyen los ambientes de software.
- d) La cualidad de software visibilidad refiere a la relación entre las clases del software, es decir, si una clase conoce a otras; contabilizando estas clases se define el grado de visibilidad.

19. En el contexto de la metodología ágil SCRUM existen tres roles (*Scrum Master*, *Product Owner*, *Developer/Desarrollador*):

- I. El *product owner* es el encargado de administrar la lista de objetivos/requisitos.
- II. El equipo de desarrolladores realiza las tareas que el *Scrum Master* selecciona.
- III. El óptimo de cantidad de desarrolladores es entre 3 y 9 personas.
- IV. El *Scrum Master* tiene por objetivo asegurarse que el equipo entienda y utilice Scrum.
- V. En caso de que el equipo no esté logrando los resultados de negocio buscados, el principal responsable será el *Scrum Master*, ya que cumple un rol de gestión.

Seleccione la opción correcta:

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

20. Con respecto a las revisiones e inspecciones:

- I. Como parte de la revisión se debe chequear la consistencia y completitud de los objetos que estén bajo revisión ya sean documentos o código. También es importante verificar que se sigan los estándares de calidad definidos.
- II. La documentación de los hallazgos de la revisión es opcional, ya que lleva un tiempo considerable y lo más importante es detectar y corregir los posibles defectos cuanto antes.
- III. Las inspecciones de código son un buen método para evaluar a los desarrolladores.
- IV. El proceso de revisión por lo general está dividido en 3 fases. En primer lugar se tiene una fase preparatoria en la cual, entre otras cosas, se conforma el equipo de revisión, se definen las fechas de la revisión y se entregan los objetos a revisar. Luego se realiza la reunión de revisión. Finalmente se cuenta con una fase pos-revisión en la que se abordan los defectos encontrados en la revisión y se realizan los cambios que se consideran necesarios.
- V. Las revisiones son más fáciles de adoptar cuando se usan metodologías ágiles que cuando se usan metodologías tradicionales (basadas en planes).

Seleccione la opción correcta:

- a) Todas son verdaderas.
- b) Tres opciones son verdaderas.
- c) Sólo (I) es verdadera.
- d) Sólo (I) y (IV) son verdaderas.

21. El factor que más influye en la estimación del esfuerzo de desarrollo de software es:
- a) el tipo de software
 - b) el tamaño
 - c) los factores del personal
 - d) el lenguaje de programación
22. Acerca de la gestión de los requerimientos puede decirse que:
- a) Incluye actividades para mantener los requerimientos actualizados, íntegros y correctos.
 - b) (a) y dentro de las actividades principales se tiene: control de versión, control de cambios, seguimiento del estado de los requerimientos y, fundamentalmente, la verificación de la implementación de los requerimientos
 - c) (a) y las actividades de control de cambios comprenden: propuesta de cambio, análisis de impacto, toma de decisión, actualización de requerimientos y actualización de planes, entre otras.
 - d) La gestión de requerimientos no requiere control de versiones, pero es necesario controlar todos los elementos que están asociados a un requerimiento para poder propagar los cambios.
23. 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 normalmente no son utilizadas como técnicas de validación.
 - d. (b) y permiten tener en cuenta el ambiente donde será ejecutado el software.
24. Respecto a requisitos:
- a) El documento de requisitos de software (SRS) debe incluir una definición de los requisitos del usuario y especificación de los del sistema.
 - b) a) y algunas características deseable en una buena especificación son: correcta, no ambigua, verificable y trazable.
 - c) b) y se debe indicar la forma en que se debe hacer y no solo el qué.
 - d) c) y algunas técnicas de validación de requisitos son: prototipado, generación de casos de prueba, revisión.
25. En el proceso de verificación y validación:
- a) Según Sommerville se busca comprobar que el sistema cumple con los requerimientos especificados (funcionales y no funcionales) , y comprobar que el software hace lo que el usuario espera.
 - b) (a) y el objetivo de las pruebas es detectar fallas en el sistema o incumplimientos con lo especificado
 - c) (b) y cuánto antes se detecten las fallas menos costoso será corregirlas
 - d) (a) y un proceso de prueba exitoso demuestra la ausencia de fallas en el software
26. Con respecto a la gestión de la configuración:
- I. Al momento de analizar una solicitud de cambio es importante considerar: las consecuencias de no realizarlo, los beneficios del cambio, el número de usuarios afectados por el cambio, el costo de realizar el cambio y el ciclo de liberación del cambio.
 - II. El grado de formalismo de la gestión de la configuración se corresponde al número de objetos comprendidos en la misma.
 - III. En el control de versiones, el método de “*File Locking*” se basa en la reserva de recursos de forma explícita cuando sabemos que vamos a modificarlos. En este caso, hasta que el usuario que está utilizando el archivo no lo libere, el resto de los usuarios accede al mismo solamente en modo lectura.
 - IV. El proceso de gestión de cambios comienza cuando un cliente completa y envía una solicitud de cambio requerida para el sistema. Esta solicitud de cambio podría contener el reporte de un error o el pedido de una nueva funcionalidad o la modificación en el comportamiento de un componente del sistema.
 - V. Hoy en día generalmente se utilizan herramientas (software) que sirven como soporte para la gestión de cambios.
- Seleccione la opción correcta:
- a) Todas son verdaderas.
 - b) Sólo I, III, IV y V son verdaderas.
 - c) Sólo I, II, III y V son verdaderas.
 - d) Sólo II, III, IV son verdaderas.

27. Sobre los casos de prueba a partir de casos de uso

- a. Una prueba derivada de un caso de uso es una prueba a nivel de sistema
- b. (a) y representa la ejecución de un escenario posible del caso de uso del cual se deriva
- c. (b) y se trata de un enfoque de caja blanca ya que es necesario contar con el código fuente para elaborar el caso de prueba
- d. (c) y que las pruebas de los flujos principales de todos los casos de uso sean satisfactorias es un buen criterio para finalizar la etapa de pruebas.

28. Con respecto a los modelos de procesos:

- I. Tanto el RUP como el MUM están centrados en la arquitectura.
- II. En la Fase de Construcción del MUM, el equipo se dedica exclusivamente a la implementación ya habiendo culminado previamente (en las fases previas) de definir de forma completa los requisitos y el diseño.
- III. Los modelos de procesos son una representación simplificada de los procesos de software. O sea, que se podría pensar que son *frameworks* que pueden ser extendidos y adaptados para crear procesos de software más específicos que se adapten mejor a la particularidad de cada proyecto.
- IV. La idea del desarrollo incremental surge a partir de la creación de los métodos ágiles. A su vez, en los métodos ágiles sólo se identifican con antelación los primeros incrementos y los restantes incrementos dependen del progreso del proyecto y de las prioridades fijadas por el cliente.
- V. El cambio es algo inevitable en todos los proyectos de software de gran escala, generando, generalmente, retrabajo. Una de las estrategias para evitar este, es que el proceso de software utilizado sea «tolerante al cambio», permitiendo que los cambios se realicen a un costo relativamente bajo.

Seleccione la opción correcta:

- a) Todas son verdaderas.
- b) Sólo I, II y V son verdaderas.
- c) Sólo III, IV y V son verdaderas.
- d) Sólo I, III y V son verdaderas.

29. Sobre los conceptos claves para un diseño de calidad:

- I. El grado de acoplamiento entre módulos depende de cuánta información se necesita sobre el otro módulo para entender el analizado. Y cuanto menor sea, mejor.
- II. La cohesión refiere a conocer porqué los elementos de un módulo están juntos.
- III. A mayor cohesión mejor será el diseño.
- IV. El principio de abierto-cerrado indica que los módulos deben ser abiertos para la extensión y modificación.
- V. La modularización y abstracción son conceptos fundamentales para construir un buen diseño.

Seleccione la opción correcta:

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

30. Sobre pruebas de requisitos no funcionales

- a. Probar los requisitos no funcionales no es uno de los objetivos de la verificación.
- b. (a) pero sí lo es de la validación.
- c. La prueba de documentación consiste en inspeccionar la documentación entregada al cliente
- d. (c) y para el caso de las pruebas de estrés es importante considerar las características del ambiente de producción