

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.
- Ante dos opciones correctas en una pregunta, se debe seleccionar la opción más completa.
- La evaluación es de carácter individual y la duración es de 2 horas.
- **El puntaje total del examen es 100 puntos y se aprueba con 60 o más puntos.**

Múltiple Opción

1. Sobre los beneficios del desarrollo incremental, seleccione la opción correcta:
 - a) Simplifica la gestión de la trazabilidad de los documentos y artefactos de software.
 - b) Hay menos probabilidad de retrabajo relacionado a cambios en los requisitos en comparación con el modelo en cascada.
 - c) El cliente usa el software antes que con un modelo de integración y configuración.
 - d) Reduce el costo total de cualquier proyecto.

La opción a) es incorrecta porque ante múltiples versiones del software (y por ende de sus documentos y artefactos), mantener la trazabilidad entre ellos resulta más costoso.

La opción c) es incorrecta porque nada indica que utilizando integración y configuración se demoren las entregas al cliente, incluso si lo que se requiere es solamente configurar un software de tercero, podría estar liberado al uso mucho antes que con un modelo incremental sin integración y configuración (cabe aclarar que estos modelos se pueden aplicar en conjunto).

La opción d) es incorrecta ya que cuando se tiene un conjunto de requisitos estables y bien definidos, con muy poca probabilidad a los cambios, un modelo en cascada por ejemplo es menos costoso que un modelo incremental.

2. Usted es el gerente de una organización que desarrolla y vende un producto de e-commerce para pequeñas empresas. Si bien el producto genérico está ya desarrollado y lo que se hace es configurarlo y adaptarlo al cliente que lo adquiere, también hay lugar para hacer pequeños ajustes o desarrollos a medida para cada cliente (que se presupuestan aparte). De acuerdo con esta realidad, seleccione la opción correcta:
 - a) Por cada cliente se debe tener un proceso de desarrollo diferente, para poder atender adecuadamente los requerimientos a medida de cada uno.
 - b) El enfoque de proceso a utilizar no puede ser ágil, ya que es necesario tener un contrato con el alcance definido para cada cliente.
 - c) Lo ideal es tener un proceso basado en el modelo de integración y configuración, y no en el modelo incremental.
 - d) Ninguna opción es correcta.

La opción a) es incorrecta porque tener clientes diferentes (con requisitos diferentes) no necesariamente implica tener un proceso diferente con cada cliente. El proceso no depende de los requerimientos particulares, sino de la dinámica de desarrollo del producto y de la relación con el cliente.

La opción b) es incorrecta porque nada indica que se deba tener un contrato con alcance definido. Se puede tener un contrato con el cliente por el producto base genérico y luego gestionar los cambios aparte, a medida que éstos vayan surgiendo.

La opción c) es incorrecta ya que lo "ideal" no puede ser determinado a partir de los datos de la letra, por ejemplo, no se sabe con qué sistemas se integran ni la complejidad de integración con estos. Por otro lado, el tener un proceso basado en el modelo incremental (además de integración y configuración) podría ser adecuado para gestionar los desarrollos a medida de cada cliente.

3. Indique la opción que describe correctamente características de Scrum.
 - a) Comprende reuniones denominadas *daily scrum meetings*, una instancia de planificación y otra de revisión por iteración. Además, se genera un incremento de software potencialmente entregable cada 4 sprints.
 - b) Está orientado a la gestión del desarrollo ágil, donde existe la figura del scrum master y cada incremento se denomina sprint. Un sprint no debería ser mayor a 4 semanas.
 - c) Incorpora los roles de scrum master y product owner. El elemento principal es el product backlog que representa todas las tareas refinadas a construir en la iteración en curso y fueron elegidas por el scrum master.
 - d) Visualiza todo el proceso como una sola fase de construcción.

La opción a) es incorrecta porque el incremento potencialmente entregable es idealmente generado en cada sprint.

La opción c) es incorrecta porque el product backlog es priorizado por el product owner. Además, el product backlog es la lista de ítems y tareas identificados, definidos y priorizados, pero no corresponde a las tareas de una iteración en curso específico, ese sería el sprint backlog.

La opción d) es incorrecta porque comprende tres fases: inicial, ciclo de sprints y clausura.

4. Sobre XP:

- a) Utiliza escenarios o historias de usuario para describir los requisitos, está dirigido por pruebas, el cliente está altamente involucrado en el equipo, utiliza programación por pares y refactorización.
- b) Utiliza escenarios o historias de usuario para describir los requisitos, el cliente está altamente involucrado en el equipo, utiliza programación por pares y refactorización, pero no está dirigido por pruebas, ya que estas se ejecutan en forma de test de regresión al finalizar el proyecto.
- c) Utiliza escenarios o historias de usuario para describir los requisitos, está dirigido por pruebas, utiliza programación por pares, pero el cliente solo está presente en las review.
- d) El cliente está altamente involucrado en el equipo, se utiliza refactorización, pero no está dirigido por pruebas ni se utilizan historias de usuario.

La opción b) es incorrecta por que sí está dirigido por pruebas.

La opción c) es incorrecta porque el cliente está muy involucrado con el equipo.

La opción d) es incorrecta por que sí está dirigido por pruebas y se utilizan historias.

5. Indique la opción correcta acerca de la mejora de procesos.

- a) El enfoque ágil se centra en mejorar procesos y la gestión de proyectos introduciendo buenas prácticas de ingeniería de software.
- b) El proceso se define durante la recolección de las mediciones y, por su parte, son las medidas quienes dirigen y guían las mejoras.
- c) El enfoque de madurez de procesos se centra en el desarrollo iterativo y en la reducción de los costos generales del proceso de desarrollo.
- d) Implica entender los procesos existentes y cambiar dichos procesos para mejorar la calidad del producto y/o reducir costos y/o acelerar los tiempos de desarrollo.

La opción a) es incorrecta porque esta es la definición del enfoque de madurez de procesos.

La opción b) es incorrecta porque el proceso debe estar definido previamente y, además, las mejoras son guiadas por los objetivos de la organización y no por las mediciones.

La opción c) es incorrecta porque esta es la definición de enfoque ágil

6. ¿Qué actividades del proceso de ingeniería de requisitos son comunes a todos los proyectos de desarrollo de software?

- I. Gestión de los requisitos.
- II. Prototipado.
- III. Relevamiento de requisitos.
- IV. Estudio de factibilidad.
- V. Especificación de casos de uso y de historias de usuario.
- VI. Análisis de interfaz de usuarios.
- VII. Validación de requisitos.

Seleccione la opción correcta:

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

La afirmación (I) es correcta, la gestión de los requisitos es una actividad genérica que se realiza en todos los proyectos de desarrollo de software.

La afirmación (II) es incorrecta, el prototipado no siempre es requerido en el proceso de requisitos.

La afirmación (III) es correcta, el relevamiento o elicitación de requisitos es una actividad que se realiza en todos los proyectos de desarrollo de software, de diversas formas.

La afirmación (IV) es correcta la factibilidad del desarrollo de los requisitos es parte del proceso de ingeniería de requisitos

La afirmación (V) es incorrecta, si bien la especificación de los requisitos es una actividad común, las técnicas de casos de uso y de historias de usuario no siempre se utilizan

La afirmación (VI) es incorrecta, puede ser que el sistema que tengamos que construir no tenga interfaz de usuario y por lo tanto esta actividad no aplique.

La afirmación (VII) es correcta, la validación de requisitos es una actividad común a todos los procesos de ingeniería de requisitos.

7. Sobre las técnicas de modelado de requisitos, seleccione la opción INCORRECTA:
- El modelo de casos de uso es un modelo de interacción.
 - Los diagramas de estado son modelos de datos.
 - Los diagramas de clase son un tipo de modelo estructural.
 - No hay ninguna opción incorrecta.

La opción b) es incorrecta porque los diagramas de estado son modelos de comportamiento.

8. Como parte de un sistema a construir, uno de los requisitos a implementar es la notificación mensual por correo electrónico de un listado con todos los pagos realizados por tarjeta de crédito al administrador del sistema. Seleccione la opción correcta:
- Dada las características de este requisito, sería recomendable modelarlo como un caso de uso y el actor es el administrador del sistema, ya que es quien inicia el caso de uso.
 - Dada las características de este requisito, sería recomendable modelarlo como un caso de uso y el actor es el reloj del sistema, ya que es quien inicia el caso de uso.
 - Dada las características de este requisito, sería recomendable modelarlo como un caso de uso y como flujo alternativo sería bueno incluir el caso en que el servidor de correo no esté disponible.
 - Ninguna opción es correcta.

Ninguna opción es correcta ya que, por definición, un actor es una entidad externa que interactúa con el sistema. Si bien el tiempo (reloj del sistema) es una entidad externa, no interactúa con el mismo dado que por sí solo no se encuentra interesado en ejecutar alguna funcionalidad, sino que, al contrario, el sistema necesitaría del tiempo para este tipo de funcionalidades. Por esta razón, no es conveniente modelar este problema con casos de uso. Lo conveniente es modelar estos casos utilizando, por ejemplo, una máquina de estado, contratos de software, especificación en lenguaje natural, entre otras.

Ref: https://eva.fing.edu.uy/pluginfile.php/283899/mod_label/intro/Is%20the%20clock%20an%20actor%3F.pdf

9. La empresa donde trabaja está encargada de desarrollar una aplicación que integra aplicaciones de terceros con diversas tecnologías. ¿Qué tipo de prototipo le parece más adecuado realizar para evaluar la factibilidad de la arquitectura propuesta?
- Cualquier prototipo horizontal.
 - Pruebas de concepto.
 - Mock-ups.
 - Ninguna de las opciones es adecuada.

Salvo la opción correcta, el resto de las opciones son incorrectas porque los prototipos horizontales tienen como objetivo mostrar muchas funcionalidades, pero no realizan ningún trabajo útil, sólo lucen como si lo hicieran (por ejemplo: Mock-ups).

10. ¿Cuál de las siguientes afirmaciones es INCORRECTA sobre requisitos?
- Si el cliente afirma que necesita contar con la aplicación en 2 meses, esta solicitud es una restricción del proyecto.
 - La técnica observación puede ser silenciosa o interactiva, se debe tener en cuenta que las personas pueden cambiar su comportamiento al sentirse observadas.
 - El proceso de ingeniería de requisitos termina cuando el "Documento de Especificación de Requisitos del Software" está validado por el cliente y en línea base.
 - Los requisitos funcionales describen cómo el sistema debería reaccionar a entradas particulares y cómo debería comportarse en situaciones particulares. Además, pueden describir lo que el sistema no debe hacer.

La opción c) es incorrecta porque el proceso de requisitos se mantiene durante todo el proyecto, incluyendo la gestión de cambios.

11. En relación con principios y estrategias de diseño, seleccione la opción correcta.
- Un buen diseño, cuanto más abstracto y genérico, permite simplificar su implementación.
 - La estrategia de "dividir y conquistar" no suele incorporar mayor complejidad al diseño, por tanto, es conveniente aplicarla hasta llegar a un problema indivisible.
 - En un diseño modular, un cambio en un componente puede implicar cambios en componentes relacionados.
 - Ante un sistema de información complejo, se recomienda implementar la mayor cantidad de patrones de diseño con el objetivo de simplificar su solución.

La opción a) es incorrecta porque un diseño abstracto y genérico, por ejemplo orientado a la reutilización, suele complejizar la implementación ya que suele requerir considerar casos o utilizar técnicas específicas.

La opción b) es incorrecta porque la estrategia "dividir y conquistar" es efectiva siempre que el costo de dividir el problema sea menor su implementación. En la medida que esto no se cumpla, no es eficiente continuar simplificando.
La opción c) es correcta porque el diseño modular requiere la comunicación entre los distintos componentes, por tanto, por ejemplo, un cambio en la interfaz de uno de ellos impacta en quienes lo utilizan.
La opción d) es incorrecta porque los patrones de diseño pueden afectarse entre sí, por lo que los beneficios de uno pueden afectar a otro, afectando la solución general del sistema..

12. La empresa ACME lo ha contratado para ser el arquitecto principal de un proyecto de desarrollo de software, por lo que:
- Será relevante que pueda entender los requisitos no funcionales del proyecto, ya que son claves para la arquitectura y diseño del sistema.
 - Su mayor responsabilidad será la gestión de equipo de desarrollo, ya que son quienes implementarán la arquitectura definida.
 - Considerando que el proyecto se ejecuta utilizando Scrum, será responsable por las estimaciones del equipo de desarrollo.
 - Considerando que el proyecto se ejecuta utilizando Scrum, deberá planificar y dirigir las reuniones diarias de seguimiento (Daily meetings).

Tal como se vio en el curso, los RNF tienen una alta relevancia para la definición de la arquitectura.

La opción b) es incorrecta porque un arquitecto de software no tiene como cometido principal la gestión del equipo de desarrollo, esta tarea suele ser llevada adelante por un jefe de proyecto o por el propio equipo (en caso de ser autogestionados)

Las opciones c) y d) son incorrectas porque Scrum no es prescriptivo en cuanto a las tareas a realizar para quien tome un rol de arquitecto.

13. Con respecto a la frase: *"El diseño de software es una actividad creativa donde se identifican los componentes del software y sus relaciones, con base en los requerimientos de un cliente"*:
- Es falsa. El diseño de software es una actividad se puede ejecutar "siguiendo una receta" que especifica actividades, precedencias y duración de cada una de ellas.
 - Es falsa. Los componentes de software se identifican al momento de la codificación, ya que es ahí que se tiene la arquitectura definida y aprobada.
 - Es falsa. La definición no considera los distintos tipos de sistemas que se pueden diseñar (aplicaciones en línea, sistemas distribuidos, juegos, etc.).
 - Es verdadera. A su vez, es relevante definir las interfaces de dichos componentes, ya que será fundamental para trabajar en sus relaciones.

La frase es verdadera. El diseño de software no puede realizarse siguiendo actividades pre definidas en tiempo y duración. A su vez, tal como indica el enunciado, la identificación de componentes es parte del proceso de diseño de software, el cuál para su definición general, no tiene por qué considerar el tipo de sistema específico.

14. Sobre fundamentos de la construcción de software, seleccione la opción INCORRECTA:
- La anticipación al cambio es un principio de construcción que se utiliza también en diseño.
 - El reuso puede afectar negativamente al interés de minimizar la complejidad.
 - La construcción para la verificación implica definir pruebas unitarias a la par que se desarrolla.
 - Los estándares utilizados para la codificación no tienen por qué ser válidos fuera de una organización.

La opción c) es incorrecta porque si bien la construcción para la verificación implica diversas técnicas para facilitar encontrar fallas en el software, no necesariamente implica definir pruebas unitarias en paralelo al desarrollo.

15. Dadas las siguientes afirmaciones sobre construcción:
- La construcción involucra actividades de diseño.
 - Los lenguajes de programación no son los únicos lenguajes considerados de construcción.
 - La codificación implica también la estructura del código y su documentación.
 - La construcción involucra actividades de verificación.
 - La calidad del código no se enfoca exclusivamente en su corrección funcional.

Seleccione la opción correcta:

- Solo las opciones I, IV y V son correctas.
- Solo las opciones I, II y III son correctas.
- Solo las opciones II, III, y V son correctas.
- Todas las opciones son correctas.

16. Sobre las revisiones de código, seleccione la opción correcta:

- a) Las revisiones de código detectan defectos en el código, no detectan fallas en el software.
- b) Conviene que las realice una única persona, y no debe participar quien codificó lo que se revisa.
- c) No son capaces de detectar problemas de sintaxis, para eso se usa el compilador.
- d) Dos opciones son correctas.

La opción b) es incorrecta ya que puede participar más de una persona en las revisiones de código, dependiendo de lo que se esté revisando, incluso personas con diferentes perfiles/conocimientos pueden ser necesarias (por ejemplo: una con enfoque de seguridad y otra con enfoque de cumplimiento de normas/estándares).

La opción c) es incorrecta porque las revisiones también pueden detectar problemas de sintaxis, no necesitan que el código "compile" e incluso se pueden realizar sobre código incompleto.

La opción d) es incorrecta porque solamente la opción a) es correcta.

17. Dadas las siguientes afirmaciones sobre los objetos de prueba, ambientes de prueba y el perfil de quien realiza las pruebas:

- I. Para realizar pruebas unitarias puedo utilizar el ambiente de desarrollo.
- II. Tanto en las pruebas unitarias como en las de integración, el objeto de prueba puede ser un conjunto de componentes.
- III. Como en las pruebas de integración quien ejecuta las pruebas es el equipo de desarrollo, necesariamente el ambiente de pruebas es el de desarrollo.
- IV. El ambiente de producción podría no ser utilizado en ninguno de los niveles de prueba.
- V. Tanto en las pruebas de sistema como en las de aceptación, el objeto, ambiente de prueba y perfil de quien realiza las pruebas, son las mismas.

Seleccione la opción correcta:

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

La afirmación (I) es correcta, en general las pruebas unitarias se realizan en el ambiente de desarrollo, aunque también se pueden realizar en otros ambientes (testing por ejemplo).

La afirmación (II) es correcta. Para las pruebas de integración es trivial la afirmación ya que requieren de al menos dos componentes que interactúan. En el caso de las pruebas unitarias, se podría llegar a probar un conjunto de componentes.

La afirmación (III) es incorrecta porque podría ser un ambiente de testing.

La afirmación (IV) es correcta ya que tanto las pruebas de sistema como de aceptación podrían ser realizadas en un ambiente de preproducción y no de producción.

La afirmación (V) es incorrecta porque en el caso de las pruebas de aceptación deben participar los clientes finales y/o usuarios, no así en las pruebas de sistema.

18. Sobre la generación de casos de prueba (CP) desde casos de uso e historias de usuario, seleccione la opción INCORRECTA:

- a) Un escenario de prueba de un caso de uso describe una ejecución completa de dicho caso de uso.
- b) Hay escenarios que pueden no estar explícitos en una historia de usuario, ya que éstas no son un método detallado de descripción de requisitos.
- c) Al momento de aplicar la técnica de generación de CP a partir de casos de uso se deben identificar todas las posibles combinaciones de flujo principal y alternativos, más allá de los casos de prueba que finalmente se terminen definiendo.
- d) Se debe definir solo un caso de prueba por cada escenario identificado a partir de un criterio de aceptación, ya que la ejecución de la prueba para valores diferentes es la misma.

La opción d) es incorrecta ya que podría haber más de un caso de prueba por cada escenario, por ejemplo, para probar diferentes casos de borde.

19. Durante un desarrollo de software se desarrolló un componente de reconocimiento de imágenes que debe cumplir con ciertos tiempos de respuesta y determinados niveles de confianza en el reconocimiento. Indique la opción correcta:

- a) Es necesario realizar pruebas no funcionales para medir el cumplimiento de los tiempos de respuesta, para lo cual se pueden usar pruebas funcionales ya definidas.
- b) En este caso, no es posible asegurar los requisitos no funcionales por medio del análisis estático.
- c) Es necesario tener en cuenta el software y hardware de base para verificar este tipo de requisitos.
- d) Todas las opciones son correctas.

20. ¿Qué elementos componen un caso de prueba?:
- I. Objeto de prueba y criterios de aceptación.
 - II. Datos de prueba y flujos alternativos.
 - III. Objetivo de prueba y resultado esperado.
 - IV. Contexto del sistema (precondiciones).

Seleccione la opción correcta:

- a) Solo las opciones I y II.
- b) Solo la opción III es correcta.
- c) Solo la opción III y IV son correctas.
- d) Todas las opciones son correctas.

La afirmación (I) es incorrecta porque los criterios de aceptación no forman parte de un caso de prueba, forman parte de las historias de usuario. Sí forma parte del caso de prueba el objeto de prueba.

La afirmación (II) es incorrecta porque los flujos alternativos no forman parte de un caso de prueba, forman parte de los casos de uso. Sí forma parte del caso de prueba el objetivo de prueba.

La afirmación (III) es correcta ya que tanto datos de prueba como resultado esperado forman parte de los casos de prueba.

La afirmación (IV) también es correcta ya que el contexto del sistema (por ejemplo: en qué pantalla está el usuario, si se tiene o no conexión a internet o a sistemas externos, etc) forma parte de los casos de prueba.

21. Suele decirse que "el mantenimiento de sistemas heredados tiene dificultades y es costoso". Seleccione la opción correcta.
- a) Esto puede deberse a que la tecnología ha quedado obsoleta, por lo que escasean las personas con el conocimiento necesario para su mantenimiento y evolución.
 - b) Esta afirmación es falsa; el contar con documentación de arquitectura garantiza que los cambios que se ejecuten no degradarán el sistema.
 - c) Esta afirmación dependerá siempre del contexto. Por ejemplo, si el mantenimiento del sistema lo realiza el mismo equipo que lo desarrolló hará que el costo de mantenerlo sea mínimo.
 - d) Una acción que permite bajar el costo de su mantenimiento es trabajar con metodologías ágiles.

La opción b) es incorrecta porque la afirmación es verdadera, ya que contar con documentación no garantiza, en ningún caso, que los cambios en evolución no degraden el software.

La opción c) es incorrecta porque, si bien muchas veces en Ingeniería de Software algunas afirmaciones dependen del contexto, mantener el mismo equipo de desarrollo para el mantenimiento no garantiza costos mínimos. Esto estará mucho más relacionado a los cambios en el contexto operativo, así como la calidad del código generado durante el proyecto.

La opción d) es incorrecta ya que las metodologías ágiles no disminuyen el costo de mantenimiento.

22. Para evaluar la mantenibilidad de un sistema podrían utilizarse las siguientes métricas:
- a) Cantidad de diagramas del documento de arquitectura y tiempo promedio en resolución de incidentes.
 - b) Cantidad de solicitudes de mantenimiento correctivo y cantidad de solicitudes rechazadas por no cumplir con el procedimiento de cambio definido.
 - c) Tiempo promedio en análisis de solicitudes de mantenimiento correctivo.
 - d) Tamaño del equipo de mantenimiento y cantidad de cambios urgentes que son rechazados.

La opción a) es incorrecta porque la cantidad de diagramas no es una métrica que permita evaluar la mantenibilidad del sistema. En todo caso, podría apuntarse a su calidad como un indicador que de pautas para evaluar la mantenibilidad.

Las opciones b) y d) son incorrectas porque, tanto las solicitudes como cambios urgentes rechazados, apuntan a medir la calidad del proceso de mantenimiento, pero no la mantenibilidad del sistema de información.

23. Sobre estrategias de adopción:
- a) Una estrategia Big-bang permite una rápida planificación y salida al mercado.
 - b) Estrategias paulatinas permiten bajar el costo de entrenamiento de usuarios.
 - c) Estrategias híbridas pueden ser adecuadas para adaptar el proceso de liberación a las necesidades del cliente con mayor flexibilidad.
 - d) El procesamiento en paralelo siempre es menos costoso ya que reduce riesgos.

La opción a) es incorrecta porque la estrategia BigBang requiere una mayor planificación, debido a que incorpora mayor complejidad.

La opción b) es incorrecta porque nada indica que un proceso de adopción paulatino tenga una correlación en los costos de entrenamiento.

La opción d) es incorrecta porque no puede afirmarse que SIEMPRE sea menos costoso, dependerá de otros factores del proyecto y/o producto.

24. Todo artefacto relacionado al proyecto de software que ha sido puesto bajo control de configuración se denomina:
- Versión.
 - Ítem de configuración.
 - Línea base.
 - Liberación (release).

La opción a) es incorrecta porque Versión es una instancia de un ítem de configuración que difiere (de alguna manera) de otras instancias del mismo ítem.

La opción c) es incorrecta porque Baseline (línea base) es un conjunto de versiones de ítems de configuración que han sido establecidos.

La opción d) es incorrecta porque Release (liberación) es una versión del sistema que ha sido liberada a los clientes para su uso.

25. Acerca del control de versiones puede decirse que:
- El control de versiones centralizado tiene un enfoque de manejo de versiones peer-to-peer.
 - En el enfoque centralizado hay menor riesgo de pérdida de datos en comparación con el enfoque distribuido.
 - Los sistemas de control de versiones almacenan la última versión de los componentes y comprenden la creación del ejecutable del sistema.
 - El control de versiones distribuido tiene un enfoque de manejo de versiones peer-to-peer, donde las computadoras se sincronizan entre sí.

La opción a) es incorrecta porque en el control centralizado contiene todas las versiones de los componentes de software y, además, el manejo peer-to-peer es una propiedad de el control de versiones distribuido.

La opción b) es incorrecta porque es el enfoque distribuido quien tiene menor riesgo de pérdida de datos.

La opción c) es incorrecta porque estos sistemas almacenan también las versiones históricas de los componentes y, por su parte, la creación del ejecutable es parte del proceso de armado (o build) y no es responsabilidad del sistema de control de versiones.

26. Respecto a los riesgos, seleccione la opción correcta:
- Si un riesgo tiene probabilidad de ocurrencia baja se ignora.
 - La identificación de riesgos es responsabilidad del gerente del proyecto.
 - Un problema es un riesgo severo.
 - Ninguna opción es correcta.

La opción a) es incorrecta dado que cada organización tiene distintos grados de tolerancia al riesgo, además que tenga una probabilidad de ocurrencia baja no significa que su impacto sea bajo y por consecuencia su severidad.

La opción b) es incorrecta porque la responsabilidad de identificar los riesgos es de todo el equipo y no solo del gerente del proyecto. El gerente puede tener visión sobre algunos riesgos, pero el equipo tiene una visión completa sobre otro tipo de riesgos.

La opción c) es incorrecta porque un riesgo negativo del proyecto que se ha materializado se considera un problema.

27. Respecto a la gestión de proyectos, en una semana X el proyecto está atrasado y está costando más de lo planificado. Seleccione la opción que cumpla con esta situación:
Aclaración: VG (Valor ganado), CA (Costo actual), VP (valor planificado)
- $VG < CA < VP$
 - $SPI = 0,8$ y $VG < VP$
 - $VP < CA < VG$
 - $CPI = 0,8$ y $SPI = 1,1$

Las opciones b), c) y d) son incorrectas dado que, si el proyecto está atrasado, tenemos que el $VG < VP$ ($SPI < 1$) y dado que el proyecto está costando más de lo planificado, tenemos que el $VG < CA$ ($CPI < 1$)

28. Dada la siguiente imagen, seleccione la opción que cumpla con esta situación:



- a) Refleja la teoría X de McGregor.
- b) Refleja la teoría Y de McGregor
- c) Incluye características de ambas teorías (X e Y).
- d) Ninguna opción es correcta.

McGregor propone dos visiones contrapuestas sobre la conducta humana en el trabajo. Según esta teoría, la concepción que los responsables de equipos tengan sobre las personas serán las que condicionen su estilo de dirección, es decir, o veo a las personas con la teoría de la X o veo a las personas con la teoría de la Y.

Teoría X

- No les gusta su trabajo y tratarán de evitarlo
- No tienen ambición ni capacidad de resolver problemas o ser creativos
- Prefieren ser constantemente dirigidos y evitan tomar responsabilidad e iniciativas
- Están motivados por las dos necesidades más básicas de Maslow
- Son egoístas, indiferentes a las necesidades de la organización y resisten el cambio

Teoría Y

- Cumplen altas expectativas si son apropiadamente motivados y si el clima de trabajo les da apoyo
- Son creativos, imaginativos, ambiciosos y comprometidos a cumplir los objetivos organizacionales
- Son autodisciplinados, pueden autodirigirse, desean responsabilidades y las aceptan.
- Están motivados por las necesidades más altas según Maslow

29. Para construir el WBS (EDT) de un proyecto:

- a) Se debe identificar todo producto de trabajo o entregable, como, por ejemplo, el diseño de la arquitectura, reportes de inspección del software y versiones construidas.
- b) No se debe incluir lo relativo a investigar la reutilización de componentes.
- c) Se debe considerar el orden y la dependencia entre estos entregables.
- d) Se debe considerar los recursos disponibles.

La creación de la EDT consiste en subdividir los entregables de proyecto en partes más pequeñas donde cada componente tiene su identificación.

La opción b) es incorrecta porque como parte de las actividades obtención de requisitos, análisis o diseño puede corresponder analizar la reutilización de componentes existentes.

La opción c) es incorrecta porque EDT no está basada en dependencias de secuencia o tiempo entre sus componentes.

La opción d) es incorrecta porque en el EDT no se consideran la cantidad de recursos disponibles para la construcción del software.

30. Sobre la estimación del esfuerzo en un proyecto de software, seleccione la opción correcta:

- a) Debe realizarse mediante juicio de experto, ya que no es un producto tangible y lo que determina la estimación es la experiencia.
- b) Puede realizarse mediante juicio de experto, basado en un modelo histórico, o por analogía, dependiendo de las características del proyecto.
- c) Se estima mediante la técnica de puntos de función cuando los requisitos no están aún definidos.
- d) Ninguna opción es correcta.

La opción a) es incorrecta porque todas las técnicas de estimación para proyectos de software incorporan, de diferentes maneras y en diversos grados, experiencias pasadas, conocimiento del futuro y factores de ajuste.

La opción c) es incorrecta porque los puntos de función se calculan contando el número de diferentes tipos de entradas, salidas, archivos internos, consultas a base de datos e interfaces en un sistema a estimar.