

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.
- **El puntaje total del examen es 100 puntos y se aprueba con 60 o más puntos.**

Múltiple Opción

1. Sobre los prototipos, seleccione la opción correcta:
 - a) No sirve para validar requisitos, sino para verificarlos.
 - b) No se usa en procesos orientados a planes.
 - c) **Sirve para relevar requisitos.**
 - d) Sirve para realizar las pruebas de aceptación.

La opción a) es incorrecta ya que un prototipo sí sirve para validar requisitos (por ejemplo, al presentarle una maqueta al cliente con el diseño de la interfaz de usuario). La opción b) es incorrecta ya que nada indica que un prototipo no pueda ser usado en un proceso orientado a planes. La opción d) es incorrecta ya que las pruebas de aceptación deben realizarse en el producto final y en un ambiente lo más parecido posible al de producción).

2. Dadas las siguientes características de un proyecto:
 - Requisitos cambiantes y poco estables
 - Contrato por tiempo y recursos
 - Cliente flexible en la adaptación de sus requisitos
 - Cliente disponible y con buena comunicación
 - Necesidad del producto (o partes de este) lo antes posible.

¿Qué tipo de enfoque y modelo de proceso sería más conveniente utilizar?

- a) Enfoque ágil con modelo incremental.
- b) Enfoque orientados a planes con modelo incremental.
- c) **Enfoque ágil con modelo incremental y de integración y configuración.**
- d) Ninguno de los enfoques anteriores es conveniente.

La opción (a) es incorrecta porque al tener un cliente flexible ante la adaptación de sus requisitos, sería recomendable incorporar un enfoque de integración y configuración (como indica la opción (b)). La opción b) es incorrecta porque un contexto con requisitos cambiantes y poco estables sumado a un contrato por tiempo y recursos no es propicio para una metodología orientada a planes, donde es necesario tener un plan para todo el proyecto, para poder medir el avance del mismo contra éste. La opción (d) es incorrecta porque la opción c) sí resulta conveniente para las características dadas del proyecto.

3. Sobre las fases del MUM, seleccione la opción correcta:
 - a) Las fases son: inicial, elaboración, construcción y transición. Donde en la fase de construcción el objetivo es estabilizar la arquitectura.
 - b) Las fases son: kick-off, construcción y transición. Donde algunos de los objetivos de la fase de transición son implantar el producto en el ambiente del cliente y transferir el conocimiento del proyecto.
 - c) Las fases son: inicial, elaboración, construcción, transición e implantación. Donde algunos de los objetivos de la fase de implantación son implantar el producto en el ambiente del cliente y transferir el conocimiento del proyecto.
 - d) **Las fases son: inicial, elaboración, construcción y transición. Donde algunos de los objetivos de la fase inicial son identificar los requisitos y riesgos del proyecto, evaluar la viabilidad del proyecto y estimar los recursos.**

La opción (a) es incorrecta porque el objetivo de la fase de construcción es culminar la implementación del software, estabilizar la arquitectura es un objetivo de fase de elaboración. La opción (b) es incorrecta porque la fase de kick-off no pertenece al MUM, faltando la fase inicial y la de elaboración. La opción (c) es incorrecta porque la fase de implantación no es una fase del MUM y el objetivo de implantar el producto pertenece a la fase de transición.

4. Sobre los equipos autogestionados, seleccione la opción correcta:
- Cada integrante decide individualmente lo que tiene que hacer.
 - Es necesario para enfoques de proceso orientado a planes.
 - Es recomendable para cualquier proyecto, aunque a veces puede no ser conveniente cuando se quiere que el equipo obedezca al jefe en relación a la asignación de tareas.
 - Ninguna opción es correcta.

La opción (a) es incorrecta porque es el equipo (como un todo) quien toma las decisiones y genera compromisos. La opción (b) es incorrecta ya que los enfoques orientados a planes no tienen por qué requerir (necesariamente) un equipo autogestionado, si bien en algunos casos puede ser beneficioso. La opción (c) es incorrecta porque el hecho de que el equipo sea autogestionado, nada tiene que ver con “hacerle caso” (o no) al jefe.

5. Sobre el modelo incremental, seleccione la opción correcta:
- Es incompatible con integración y configuración.
 - Solo se utiliza en enfoques ágiles.
 - Facilita el hacer frente al cambio cuando se realizan entregas periódicas, pero puede generar retrabajo.
 - Ninguna opción es correcta.

La opción (a) es incorrecta porque ambos modelos (incremental e integración y configuración) pueden combinarse en un mismo proyecto. La opción (b) es incorrecta porque el enfoque incremental también puede ser usado en enfoques orientados a planes, por ejemplo el MUM. La opción (d) es incorrecta porque la opción c) es correcta.

6. Sobre ingeniería de requisitos, seleccione la opción INCORRECTA:
- Los requisitos pueden ser definidos con diferentes niveles de detalle dependiendo del contexto.
 - Un requisito del dominio se deriva del dominio de aplicación y no de las necesidades de los usuarios.
 - Los requisitos no funcionales deben poder medirse cuantitativamente para evaluar si el sistema los cumple o no.
 - Un requisito no funcional puede estar directamente vinculado con múltiples requisitos funcionales.

La opción (c) es incorrecta porque pueden existir requisitos no funcionales para los cuales no se pueda definir una métrica o se pueda asociar el valor de dicha métrica a la satisfacción de un requisito de los usuarios, ej: mantenibilidad.

7. Sobre las técnicas de obtención de requisitos, seleccione la opción correcta:
- La entrevista es efectiva para obtener requisitos organizacionales generando espacios de encuentro entre personas con diferentes niveles jerárquicos.
 - El prototipo es efectivo para comprender mejor ciertos requisitos y debe ser económico, por lo que no se hace énfasis en su robustez, ya que será descartable.
 - La tormenta de ideas es efectiva para descubrir y combinar ideas, pero es necesario una crítica constructiva y debate para no generar ideas imposibles de realizar.
 - La etnografía es efectiva porque se observa directamente a los usuarios mientras realizan sus actividades, pero se puede generar un sesgo importante dependiendo de a qué usuario se observe.

La opción (a) es incorrecta porque muchas veces los mecanismos de toma de decisiones no se corresponden con las relaciones jerárquicas teóricas, por lo que las personas tienden a evitar discutir sobre aspectos políticos y organizacionales. La opción (b) es incorrecta porque los prototipos evolutivos son una base sólida para la construcción del sistema, por lo que su robustez es vital. La opción (c) es incorrecta porque no se permite criticar ni debatir sino generar tantas ideas como sea posible, mutar y combinar ideas.

8. Dadas las siguientes características deseables de un requisito:
- Es Completo si describe de forma precisa una necesidad.
 - Es Factible si puede ser implementado en el contexto del sistema y de su ambiente operacional y considerando restricciones del proyecto.
 - Es Necesario si fue priorizado considerando a todos los interesados.
 - Es No Ambiguo si no admite múltiples interpretaciones.
 - Es Verificable si es posible realizar pruebas para determinar si el requisito se satisface.

Seleccione la opción correcta:

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

La afirmación I es incorrecta porque un requisito es Completo si contiene toda la información para entenderlo; si describe de forma precisa una necesidad, se denomina Correcto. La afirmación III es incorrecta porque un requisito es Necesario solo si tiene cierto valor para el negocio; si fue priorizado por los interesados, se denomina Priorizado.

9. Sobre aspectos a considerar para la escritura de requisitos, seleccione la opción INCORRECTA:
- Utilizar el mismo nivel de detalle para describirlos.
 - No utilizar múltiples términos para el mismo concepto.
 - Si los clientes están muy involucrados en la definición de requisitos, se puede incluir menos detalle.
 - Intentar, en lo posible, utilizar una voz activa y evitar intercalar con una voz pasiva.

La opción (a) es incorrecta porque no todos los requisitos deben tener el mismo nivel de detalle, aunque se recomienda mantener el mismo nivel en requisitos relacionados.

10. Sobre las técnicas de casos de uso e historias de usuario, seleccione la opción INCORRECTA:
- Un caso de uso provee una representación de alto nivel de los requisitos de usuario.
 - Un caso de uso incluido se ejecuta siempre, en tanto una extensión sólo si se cumple una condición particular.
 - La extensión de la redacción de las historias de usuario es variable y lo determina el equipo de desarrollo.
 - Los criterios de aceptación deben cubrir la mayor cantidad de casos posibles, tanto comunes como alternativos.

La opción (b) es incorrecta porque tanto los casos de uso incluidos como las extensiones pueden formar parte de flujos alternativos que no se ejecuten en una instancia específica del caso de uso. En caso de que efectivamente formen parte del flujo que se ejecuta, sí es válida la afirmación.

11. Sobre el estilo arquitectónico "en capas", seleccione la opción correcta:
- Se recomienda que la cantidad de capas no exceda la mitad del número total de componentes del sistema.
 - Tiene como ventaja que, manteniendo su especificación, puede modificarse su implementación sin que el resto de las capas requiera cambios.
 - Para incrementar su desempeño (performance), es recomendable incrementar la cantidad de capas. Esto permite una mayor cohesión en el sistema.
 - Siempre una capa provee servicios a su capa inmediatamente superior.

La opción (a) es incorrecta ya que la cantidad de capas de un software no se vincula directamente con su cantidad de componentes. La opción (c) es incorrecta porque el incrementar la cantidad de capas penaliza la performance del sistema. La opción (d) es incorrecta porque, en un modelo relajado, se pueden proveer servicios a diversas capas (no exclusivamente a la superior)

12. Sobre los diagramas de despliegue, seleccione la opción correcta:
- Permite identificar la infraestructura sobre la cual se ejecutan los componentes de software.
 - No es necesario utilizarlos cuando la infraestructura se despliega en la nube (cloud computing), ya que esta es virtual.
 - Dada su criticidad, es un elemento del proyecto que siempre forma parte de la línea base.
 - Es un buen elemento de comunicación para usuarios no técnicos, ya que permite entender cómo funciona el sistema desde otra perspectiva.

La opción (b) es incorrecta, el despliegue en infraestructura virtual no inhibe el desarrollo de diagramas de despliegue. La opción (c) es incorrecta, dependiendo de las características del proyecto, el artefacto puede o no ser parte de la línea base. La opción (d) es incorrecta, debido a sus características, el diagrama de despliegue no es un "buen elemento" de comunicación para usuarios no técnicos.

13. Sobre el proceso de diseñar la arquitectura, seleccione la opción correcta:
- Sigue siempre un proceso predefinido, que depende de la metodología de desarrollo utilizada en el proyecto.
 - Debe tener en cuenta requisitos funcionales y no funcionales.
 - Involucra a los usuarios finales, quienes son la principal fuente de requisitos no funcionales y atributos de calidad del sistema a construir.
 - Cuando se trata de un proyecto de sustitución de sistemas legados, el proceso de diseño se simplifica ya que se replica la arquitectura del sistema original.

La opción (a) es incorrecta porque el diseño de la arquitectura puede entenderse como una actividad creativa, no necesariamente predefinida en sus actividades. La opción (c) es incorrecta, el diseño de la arquitectura no suele involucrar usuarios finales. La opción (d) es incorrecta porque, por más que se este sustituyendo un sistema por otro, el nuevo sistema deberá definir su arquitectura en base a las características del nuevo software y las necesidades de la organización.

14. Sobre fundamentos de la construcción de software, seleccione la opción correcta:
- a) Para minimizar la complejidad hay que hacer énfasis en la escritura legible y simple de código.
 - b) Dado que el software evolucionará constantemente, es siempre necesario anticipar el cambio.
 - c) No existen técnicas específicas que promuevan la construcción para la verificación.
 - d) Dado el nivel de abstracción al cual se codifica, el reúso se da principalmente a nivel de las clases/métodos.

La opción (b) es incorrecta porque no todo software tiene por qué evolucionar, y aunque lo haga, la anticipación del cambio puede depender de necesidades identificadas a futuro en el proyecto, ej.: un requisito de extensibilidad. La opción (c) es incorrecta porque sí existen técnicas, ej.: usar estándares de codificación para facilitar las revisiones de código. La opción (d) es incorrecta porque el reúso no tiene que ver con el nivel de abstracción al cual se codifica, sino con una intención manifiesta de llevarlo adelante a diferentes niveles de abstracción.

15. Dadas las siguientes afirmaciones sobre construcción:
- I. Depende del modelo de ciclo de vida utilizado.
 - II. Involucra actividades de diseño.
 - III. Implica aspectos de gestión de actividades y asignación de recursos.
 - IV. Involucra actividades de verificación.
 - V. Puede depender del entorno de desarrollo que se utilice.

Seleccione la opción correcta:

- a) Solo las opciones I, III y IV son correctas.
 - b) Solo las opciones I, II y III son correctas.
 - c) Solo las opciones I, II, III y IV son correctas.
 - d) Todas las opciones son correctas.
16. Sobre las técnicas de prueba, seleccione la opción correcta:
- a) El testing exploratorio es una técnica basada en la experiencia y se utiliza a nivel de pruebas unitarias.
 - b) Las técnicas de caja blanca se basan en la especificación del software.
 - c) No es conveniente usar técnicas de caja blanca para las pruebas de performance.
 - d) Ninguna opción es correcta.

El testing exploratorio en general se usa a nivel de pruebas de sistema ya que explora y prueba el producto construido a través de la interfaz de usuario. Las técnicas de caja blanca se basan en el código fuente y el flujo de control (o de datos) del mismo, no se basan en la especificación. En algunos casos es muy útil utilizar pruebas de caja blanca para pruebas de performance, por ejemplo: en la conformación de consultas a la base de datos, en la ejercitación de loops o métodos recursivos observando el uso de memoria, entre otros.

17. En relación a las estrategias de integración, y al diseño y ejecución de las pruebas, seleccione la opción correcta:
- a) La estrategia de Big-bang es la más costosa de realizar.
 - b) Se debe tener en cuenta la arquitectura para planificar las pruebas.
 - c) La estrategia por disponibilidad en general no se usa ya que en general el software como un todo no está disponible al momento de las pruebas de integración.
 - d) Si las pruebas unitarias fueron exitosas, las pruebas de integración deberían ser pocas.

La opción (a) es incorrecta ya que el diseño de las pruebas con una estrategia de Big-Bang no requiere la construcción de drivers ni de stubs, por tanto es menos costosa que Bottom-Up o Top-Down. La opción (c) es incorrecta ya que la estrategia por disponibilidad se usa bastante ya que se basa en la disponibilidad de las piezas de código no bien los desarrolladores culminan las pruebas unitarias. Además, la realización de las pruebas de integración no necesitan el software como un todo para poder probarlo. La d) es incorrecta ya que la cantidad de pruebas de integración a realizar no debería disminuir (o deberían ser independientes) en base al éxito de las pruebas unitarias.

18. En relación a los criterios de entrada y salida de las pruebas, seleccione la opción INCORRECTA:
- a) Un criterio de salida adecuado podría ser “cubrir todos los flujos normales y alternativos de los casos de uso”
 - b) Un criterio de entrada adecuado podría ser que la funcionalidad a probar esté integrada en el ambiente de pruebas.
 - c) Un criterio de salida adecuado podría ser que el cubrimiento logrado sea menor al 90%.
 - d) Un criterio de entrada adecuado podría ser que el ambiente de pruebas esté disponible.

La opción (c) es incorrecta ya que el criterio no debería ser “menor” al 90%, sino “mayor” ya que sino se podrían terminar las pruebas sin siquiera haber probado.

19. Una clasificación de defectos podría basarse en:
- El impacto que el defecto genera en el cliente.
 - La causa que originó el defecto.
 - El desarrollador que introdujo el defecto.
 - Dos opciones son correctas.

Las opciones (a) y (b) son correctas. La opción (c) es incorrecta ya que identificar individualmente el desarrollador que introdujo la falla genera una exposición innecesaria del mismo ante el equipo y puede afectar su moral.

20. ¿Cuál o cuáles de los siguientes objetivos corresponden a la verificación de software?
- Revisar los productos de software para encontrar defectos.
 - Encontrar fallas en el software.
 - Corregir los defectos del software.
 - Evaluar la relación con el cliente.
 - Evaluar los atributos de calidad del software.

Seleccione la opción correcta:

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

La afirmación III es incorrecta ya que el objetivo de la verificación es detectar el defecto, la actividad de corregirlo dependerá de otra disciplina, dependiendo del tipo de defecto que sea. La opción IV es incorrecta ya que la relación con el cliente no es posible evaluarla a través de actividades de verificación.

21. Sobre las estrategias de adopción de software, podemos mencionar:
- Big-bang, por capas y procesamiento en paralelo
 - Procesamiento en paralelo y estrategias paulatinas
 - Estrategias híbridas, estrategias por capas y big-bang.
 - Estrategias ágiles, estrategias híbridas y estrategias paulatinas

La opción (a) y (c) no son correctas ya que “por capas” no es una estrategia. La opción (d) es incorrecta porque “estrategias ágiles” no es una estrategia de adopción según lo trabajado en el curso.

22. Sobre los distintos tipos de mantenimiento, seleccione la opción INCORRECTA:
- El "mantenimiento correctivo" pone foco en la reparación de fallas.
 - El "mantenimiento evolutivo" busca incorporar o evolucionar funcionalidades.
 - El "mantenimiento adaptativo" es necesario cuando algunos aspectos del entorno del sistema cambian (por ejemplo: un cambio normativo).
 - Todos los distintos tipos de mantenimiento requieren ser abordados por un equipo experto en análisis de impacto.

Las opciones (a), (b) y (c) presentan la definición correcta del tipo de mantenimiento mencionado. En cambio, la opción (d) plantea una afirmación incorrecta, ya que no suele contarse con equipos “expertos en análisis de impacto”.

23. ¿Cuál de las siguientes afirmaciones es verdadera?
- El costo del mantenimiento de un software es, por lo general, mayor al costo de su desarrollo.
 - El análisis de impacto de un cambio en software ya productivo requiere siempre de la aprobación del arquitecto de software.
 - Para mitigar riesgos en la sustitución de un sistema por otro es recomendable una estrategia de adopción tipo big-bang.
 - Para reemplazar un sistema heredado se debe utilizar una metodología ágil de desarrollo. Esto permite ir visualizando las nuevas funcionalidades de forma continua, habilitando una estrategia paulatina de sustitución.

La opción (b) es incorrecta, el proceso de análisis y aprobación no siempre debe incluir al arquitecto (podrá depender del tipo de cambio, por ejemplo). La opción (c) es incorrecta porque la estrategia Big-Bang es la que suele tener asociado el mayor nivel de riesgo. La opción (d) es incorrecta porque el reemplazo de sistemas no necesariamente implica trabajar bajo metodologías ágiles.

24. Sobre las estimaciones de esfuerzo en proyecto, seleccione la opción correcta:
- Son realizadas por gente experta, que generalmente se especializa en técnicas y procedimientos de estimación.
 - Deben ser revisadas y ajustadas durante el proyecto.
 - Deben ser realizadas con el usuario final para generar mayor compromiso y certeza.
 - Deben ser aprobadas por el cliente.

La opción (a) es incorrecta ya que suelen participar distintos roles y perfiles en las tareas de estimación, no necesariamente “expertos en técnicas de estimación”. La opción (c) es incorrecta porque el usuario final no suele participar de la estimación, esto recae en roles del equipo de desarrollo. La opción (d) es incorrecta ya que las estimaciones no “deben” ser aprobadas por el cliente.

25. Sobre la gestión del tiempo en un proyecto de software, seleccione la opción correcta:
- Esfuerzo y duración de una tarea son la misma métrica de seguimiento.
 - El esfuerzo de una tarea indica, por lo general, la cantidad de horas que insumiría completarla.
 - La duración de una tarea indica, por lo general, el tiempo que le llevaría a un programador experiente completarla.
 - Esfuerzo y duración de una tarea son determinadas por el Jefe de Proyecto (ya que es el responsable de su gestión).

La opción (a) es incorrecta, el esfuerzo se relaciona con “cantidad”, mientras que duración indica cómo esa cantidad de esfuerzo será ejecutada a lo largo del tiempo. La opción (c) es incorrecta ya que la duración no necesariamente está relacionada al tiempo que le llevaría un programador con experiencia, se podría tomar la convención que el proyecto o equipo entienda conveniente... La opción (d) es incorrecta, en múltiples metodologías, el esfuerzo, por ejemplo, es determinado por el equipo de desarrollo.

26. “Un riesgo es un evento o condición incierta que, de producirse, tiene un efecto positivo o negativo en uno o más de los objetivos del proyecto, tales como el alcance, el cronograma, el costo y la calidad.”
- La afirmación es falsa, los riesgos en los proyectos de software no tienen efectos positivos.
 - La afirmación es verdadera. A su vez, puede indicarse que un riesgo puede tener diversas causas, así como también diverso impacto.
 - La afirmación es verdadera. De forma complementaria, la gestión de riesgos debe definir un plan de contingencia para cada uno de los riesgos identificados.
 - La afirmación es falsa, los riesgos no afectan la calidad del producto. La afectación de la calidad viene dada por la introducción de bugs durante el proceso de construcción.

La afirmación es verdadera, por tanto (a) y (d) son opciones incorrectas. Por otro lado, (c) es incorrecta debido a que parte de gestionar los riesgos implica definir sobre cuáles de ellos se realizarán planes de contingencia y cuáles serán monitoreados para realizar el plan de contingencia cuando se entienda pertinente.

27. Algunas técnicas de estimación de software son:
- Analogía y Juicio de Terceros.
 - Juicio de expertos y estimación por puntos de función.
 - Análisis cuantitativo y planning poker.
 - Analogía y estimación conductual.

La opción (b) es la única que incluye técnicas válidas. El resto introduce “técnicas” que no se constituyen como reales ni vistas el curso (Juicio de Terceros, Análisis cuantitativo, estimación conductual)

28. Durante un proyecto de software debe gestionarse, de forma simultánea, su alcance, tiempos y costos, esto implica:
- A mayor tiempo, mayor costo del proyecto.
 - No se puede modificar el alcance sin evitar impactar los tiempos.
 - Cambios en recursos durante el proyecto no implica impacto en el alcance.
 - El alcance, tiempo y costo deben considerarse como atributos de calidad.

La opción (a) es incorrecta porque mayor tiempo no implica mayor costo (podría disminuir los recursos asignados al proyecto para no impactar en costos). La opción (b) es incorrecta, podría incrementarse el alcance e incorporar más recursos al proyecto, evitando así impactar en los tiempos. La opción (d) es incorrecta, no son atributos de calidad según lo planteado en el curso.

29. Sobre las actividades de gestión de la configuración, seleccione la opción INCORRECTA:
- a) Durante la gestión de la liberación se lleva un registro de las diferentes versiones liberadas para uso del cliente.
 - b) Durante la gestión de cambios se evalúa el costo e impacto de un cambio antes de implementarlo.
 - c) Durante el armado del sistema (building) se ensamblan los componentes para generar un ejecutable a liberar.
 - d) Durante la gestión de versiones se asegura que los cambios realizados en los componentes no interfieran entre si.

La opción (c) es incorrecta porque el ejecutable que se genera no necesariamente debe ser liberado para uso del cliente.

30. Sobre la gestión de cambios, seleccione la opción correcta:
- a) En ciertos contextos es posible omitir el comité de control de cambios.
 - b) Durante el desarrollo la decisión de un cambio podría ser tomada directamente por el desarrollador.
 - c) El cliente podría estar directamente involucrado en la toma de decisiones de la gestión de los cambios.
 - d) Todas las opciones son correctas.