

- Cada pregunta de la **parte múltiple opción** contestada correctamente tiene un valor de 2 puntos. Cada pregunta incorrecta de la múltiple opción resta 0,66 puntos.
- **El puntaje total del parcial es 50 puntos.**

## Múltiple Opción

- 1) La arquitectura de software
  - a. es lo primero que hay que obtener en un proyecto de desarrollo de software
  - b. **incluye (entre otros) a los subsistemas que componen el sistema software**
  - c. (b) y es normalmente sencillo cambiar esta arquitectura luego que el sistema software se ha liberado (instalado en el cliente)
  - d. (b) y durante el proceso de creación de dicha arquitectura no se deben considerar conflictos de propiedades de calidad (ejemplo: mantenibilidad y performance). Estos conflictos se resuelven luego en el diseño detallado.

---

- 2) Para crear la arquitectura de software se deben considerar los requisitos del sistema a construir. ¿Cuál de las siguientes afirmaciones es válida?
  - a. No se consideran los requisitos funcionales
  - b. No se consideran los requisitos no funcionales
  - c. Se consideran tanto los requisitos funcionales como los no funcionales, basándose fuertemente en los funcionales.
  - d. **Se consideran tanto los requisitos funcionales como los no funcionales, basándose fuertemente en los no funcionales.**

---

- 3) Los estilos arquitectónicos permiten utilizar una solución entendible y probada a determinados problemas
  - a. y definen la estructura del sistema de software
  - b. **(a) y basar una arquitectura de software en estilos conocidos permite una mejor y más rápida comprensión por parte del equipo de desarrollo de las características más relevantes de la misma.**
  - c. (b) y el estilo de tubos y filtros tiene como desventaja que las actividades de respaldos, seguridad, control de acceso y recuperación están centralizadas.
  - d. Ninguna de las anteriores

---

- 4) El estilo pizarrón se compone de un almacén de datos y de fuentes de conocimiento que manipulan datos de dicho almacén
  - a. **En el estilo pizarrón las fuentes de conocimiento responden a cambios en el pizarrón**
  - b. En el estilo pizarrón las fuentes de conocimiento no responden a los cambios en el pizarrón ya que dependen de cambios no relacionados con el almacén de datos
  - c. (a) y El estilo cliente servidor en dos niveles no se debe utilizar más ya que existe una mejor opción que es el estilo cliente servidor en múltiples niveles
  - d. (b) y El estilo cliente servidor en dos niveles no se debe utilizar más ya que existe una mejor opción que es el estilo cliente servidor en múltiples niveles

---

- 5) Acerca de la evaluación de la arquitectura de software
  - a. No debe realizarse ya que es un proceso que ha mostrado ser caro y que no resulta conveniente en cuanto a su costo-beneficio
  - b. **Es importante contar con evaluaciones técnicas de la arquitectura para conocer si la misma soporta los requisitos y las propiedades de calidad establecidas para el producto de software**
  - c. (b) y uno de los métodos de evaluación es el método Service Oriented Architecture (SOA) que posibilita evaluar la arquitectura mediante la consideración de esta como uno o varios servicios
  - d. (b) y, sin embargo, no es muy preocupante si no se realiza ya que los defectos inyectados en la arquitectura de software no son costosos de remover, incluso si estos se detectan cuando el producto ya fue liberado al cliente.

---

- 6) Acerca del diseño de la interfaz de usuario
  - a. Es importante comenzar tempranamente (en el proceso de desarrollo de software) el diseño de la interfaz de usuario
  - b. El diseño de la interfaz de usuario no debe preocupar al equipo de desarrollo ya que lo más importante es que la lógica de la aplicación funcione. Luego de obtenido el funcionamiento completo del sistema se diseñará la interfaz de usuario que sea necesaria.
  - c. **(a) y dos aspectos clave del diseño de la interfaz de usuario son considerar cómo se presentará la información al usuario y cómo el usuario interactuará con el sistema**
  - d. (b) y dos aspectos clave del diseño de la interfaz de usuario son considerar cómo se presentará la información al usuario y cómo el usuario interactuará con el sistema

- 
- 7) Acerca del diseño
- El diseño de software se define como la actividad que transforma una especificación de un caso de uso en clases de un lenguaje orientado a objetos o en programas y procedimientos en un lenguaje procedural.
  - El diseño de software consiste en usar el lenguaje UML para especificar clases y sus interacciones
  - El diseño de software es normalmente una actividad poco creativa y normalmente se usan herramientas automáticas que lo generan.
  - El diseño de software refina la arquitectura de software para obtener una solución (a nivel de diseño) de todo el sistema.**
- 
- 8) Respecto a la evaluación de los atributos de calidad de un producto de software.
- Los atributos de calidad internos de un producto de software solo se pueden evaluar una vez que el producto ya se encuentra construido y tiene comportamiento, y solo son visibles a los desarrolladores, no para los usuarios.
  - Para poder evaluar los atributos de calidad externos de un producto de software este tiene que tener comportamiento, lo que exige que ya esté – al menos parcialmente – construido.
  - (b) y un prototipo permite evaluar indirectamente algunos atributos de calidad externos.**
  - (c) y la construcción de un prototipo de un producto de software requiere utilizar las mismas herramientas que las que se van a utilizar para la construcción del producto final.
- 
- 9) Respecto a la Ingeniería de Software:
- Se trata de una disciplina con una larga tradición, completamente estabilizada y se encuentra muy próximo el momento en que sea considerada una profesión al estilo de la abogacía y la medicina.
  - Ha cambiado mucho en los últimos 20 años, entre otras razones debido al abaratamiento del hardware y el sostenido incremento en la potencia de procesamiento, el incremento en la capacidad de almacenamiento, los problemas que presenta el modelo de proceso en cascada y la presión por reducir el tiempo de salida al mercado (time to market) resultante del incremento de la competencia originado por el proceso de globalización.**
  - Consiste esencialmente en la construcción de programas, por lo que puede asimilarse al desarrollo de software.
  - A medida que el software se vuelve más importante para la vida y funcionamiento de la sociedad, son cada vez menores las exigencias que la sociedad plantea respecto al nivel de calidad del software y disminuye a la vez la importancia de las cuestiones legales y reglamentarias relacionadas con la producción y uso del software.
- 
- 10) Respecto a modelos de proceso de software:
- El proceso en cascada al día de hoy prácticamente ha dejado de ser utilizado en la industria, habiendo sido sustituido por otros modelos como el proceso en espiral.
  - Un proyecto en la industria se va a encarar siguiendo un proceso en fases con liberaciones parciales en producción en 2 fases. Una alternativa interesante para reducir los riesgos del proyecto sería encarar la primera fase siguiendo el modelo Transformacional y la fase final de acuerdo al modelo Operacional.
  - Una de las desventajas de un proceso en fases con liberaciones parciales en producción es que no permite acortar el plazo para que cliente y usuarios puedan contar con resultados que les sean de utilidad.
  - Una de las ventajas del modelo de proceso en fases con evaluaciones internas parciales es que permite acortar el plazo para contar con retroalimentación de parte de cliente y usuarios respecto a atributos de calidad externos del producto.**
- 
- 11) Dados los siguientes proyectos y modelos de proceso.
- Proyecto:
- A - de alto riesgo, muy complejo, alcance no del todo definido, varios subconjuntos de funcionalidad identificados como útiles para los usuarios, larga duración (dos años o más).
- B – Riesgo medio, alcance definido, subconjunto de funcionalidad definido y útil para los usuarios, duración media (seis meses).
- C – Riesgo bajo, simple, alcance definido, corta duración (seis semanas).
- Proceso:
- 1 – Cascada
- 2 – En Fases con Evaluaciones Parciales Internas
- 3 – En Fases con Liberaciones Parciales en Producción
- 4 – De Prototipación
- Marque la asignación que le parezca más adecuada. Si un mismo proyecto aparece asignado a más de un proceso, debe entenderse que cualquiera de los dos resultaría adecuado.
- Para los procesos 2 y 3 se puede especificar además entre paréntesis el modelo de proceso a utilizar en cada fase. Por ejemplo 2(1) significa Proceso en Fases con Evaluaciones Parciales Internas y en cada fase se aplica Cascada. 3(2(1))

---

significa que en este caso en cada fase de 3 se aplica En Fases con Evaluaciones Parciales Internas y en cada fase de 2 se aplica Cascada.

- a. A3(2(1)), B3(1), C2(1), C1
- b. A1, A4, B1, B2(3(1)), C3(2(1)), C3(1)
- c. A4, B2(1), B(1), C3(1), C1
- d. A3(2(1)), B(1), C2(1), C1

---

12) El Proceso de Ingeniería de Requisitos

- a. consiste en la obtención, análisis, especificación, verificación y validación de los requisitos del sistema y termina cuando se los tiene descriptos y validados.
- b. termina cuando se tiene una versión estable del Documento de Especificación de Requisitos del Software
- c. termina cuando el Documento de Especificación de Requisitos del Software no sólo es estable sino que está validado por el cliente y en línea base.
- d. continúa hasta el final del proyecto, porque incluye la gestión del cambio

---

13) Un actor principal:

- a. Es todo aquel que interviene en un caso de uso
- b. Es siempre el que inicia el caso de uso
- c. Obtiene un resultado de valor del caso de uso
- d. Ninguna de las anteriores

---

14) Sobre los Casos de Uso indique cuáles de las siguientes afirmaciones son correctas.

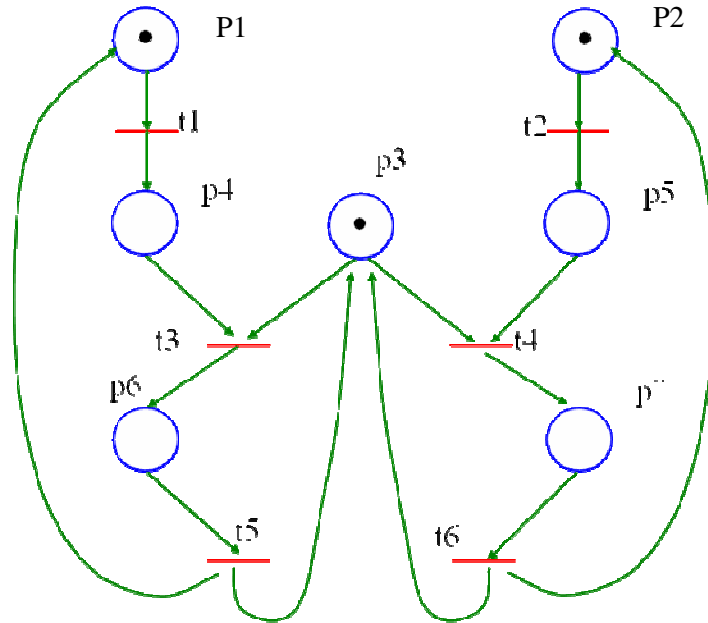
- I. Son una técnica de relevamiento y modelado de requisitos.
  - II. Describen la interacción del sistema con otros actores (se enfoca más que nada en la visión del usuario).
  - III. Cada Caso de Uso es un Requisito del sistema a construir.
  - IV. Son un modelo bastante complejo y que en general no se muestra a los usuarios.
  - V. En general, son una buena base para construir conjuntos de casos de prueba aunque no contemos aún con el sistema construido.
- a. I, II y III son correctas.
  - b. II y III son correctas.
  - c. I, II y V son correctas.
  - d. Todas son correctas menos la IV.

---

15) Cuando se modela con redes de Petri, generalmente:

- a. Las condiciones se modelan como lugares y los eventos como transiciones
  - b. Las condiciones se modelan como transiciones y los eventos como lugares
  - c. Las condiciones y los eventos se modelan como lugares y las transiciones marcan los pasajes de una condición a otra
  - d. Las condiciones y los eventos se modelan como transiciones y las marcas indican un escenario particular.
-

16) En la siguiente red de Petri, con una marca inicial de p1, p2 y p3

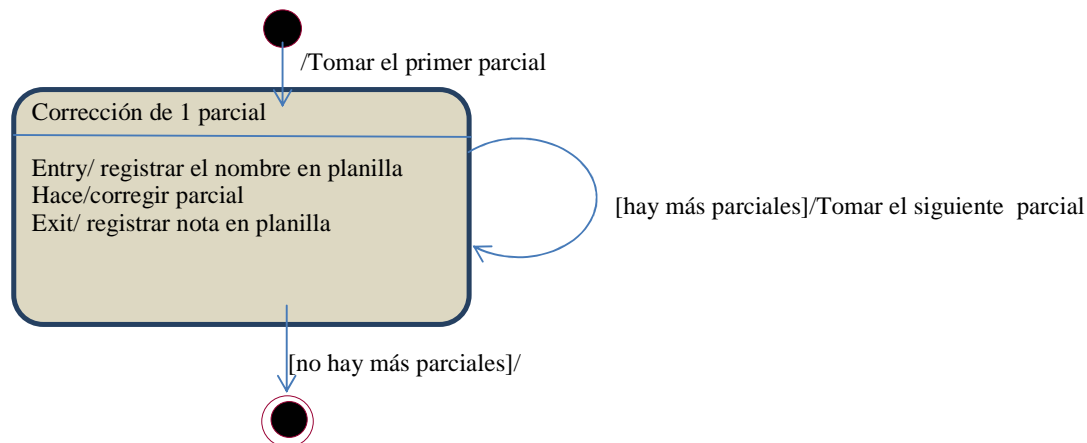


- a. La secuencia t1, t2, t3, t5, t4, t6 es posible
- b. Es seguro que en algún momento se disparará t4
- c. Existe una secuencia que entra en deadlock
- d. Ninguna de las anteriores

17) En el proceso de los requisitos, usar una tabla de decisión para modelar requisitos

- a. Permite obtener más requisitos que cualquier otra técnica
- b. Permite detectar omisiones y verificar la no ambigüedad de los requisitos
- c. Se torna impracticable si el conjunto de estados es muy extenso
- d. Debería considerar todas las combinaciones posibles de condiciones y acciones

18) En el siguiente diagrama de estados:



el orden de las acciones y actividades cuando se ejecuta la autotransición es el siguiente:

- a. registrar el nombre en planilla, corregir el parcial, registrar nota en planilla, tomo el siguiente parcial
- b. registrar nota en planilla, tomo el siguiente parcial, registrar el nombre en planilla, corregir parcial
- c. tomar el siguiente parcial, registrar el nombre en planilla, corregir parcial, registrar nota en planilla
- d. tomar el siguiente parcial, corregir parcial, registrar el nombre en planilla, registrar nota en planilla

- 
- 19) En el contexto de la planificación de las actividades de un proyecto se llama holgura total de una actividad a:
- la diferencia entre el fin temprano y el comienzo temprano de una actividad.
  - el buffer que se agrega al final del proyecto para contemplar los riesgos
  - la diferencia entre el fin tardío y el fin temprano de una actividad.
  - la diferencia entre el comienzo tardío y el comienzo temprano de una actividad
- 
- 20) Al estimar la duración de una actividad del proyecto se define que su duración más probable es 10 días, la más optimista es 6 y la pesimista es 14. Utilizando PERT, ¿cuál es la duración estimada para dicha actividad?
- 10 días.
  - 9 días.
  - 12 días
  - Falta información sobre el riesgo de dicha actividad para poder calcular su duración.
- 
- 21) En relación a la Gestión de Riesgos
- Un riesgo es un evento o condición inciertos, que, en caso de producirse, afecta todos los objetivos del proyecto.
  - La gestión de los riesgos tiene como único objetivo la disminución de la probabilidad y el impacto de los riesgos negativos.
  - Debe ser una actividad proactiva y mantenida durante todo el proyecto, en la que participa todo el equipo de desarrollo.
  - Todas las anteriores son correctas.
- 
- 22) Un estudio de factibilidad
- Tiene como objetivo determinar si vale la pena embarcarnos o continuar en un proyecto
  - en general tiene como resultado un informe de factibilidad que incluye análisis de costo/beneficio, sin aventurar recomendaciones de si seguir o no con el proyecto
  - La factibilidad técnica se estudia luego de que se inicia el proyecto, se relevan los requerimientos y se estudian las arquitecturas candidatas. Recién entonces se puede decir que el proyecto es factible técnicamente o no.
  - Todas las anteriores son correctas.
- 
- 23) En cuanto a la Gestión de Proyectos una de las siguientes afirmaciones no es correcta, indique cual.
- El Plan de Gestión del Proyecto permite comunicar entre otras cosas el alcance del proyecto, la metodología a utilizar, el calendario, la organización del equipo y los planes subsidiarios (por ej. planes de capacitación, aseguramiento de la calidad, gestión de la configuración, gestión de riesgos, etc.).
  - Algunas técnicas para el registro y control del avance del proyecto son: Diagrama de Gantt, Diagrama de Evolución de Gastos y Enfoque de Valor Ganado.
  - El diagrama de evolución de gastos permite ver lo gastado respecto a lo planificado a gastar con respecto a los logros alcanzados.
  - El enfoque de valor ganado corresponde a un modelo en el cual se asignan a todas las actividades planificadas un valor a alcanzar que coincide inicialmente con su costo planificado.
- 
- 24) Si se quiere calcular los PFSA de la siguiente función:
- Se selecciona de un menú la opción de ingreso de datos de factura de proveedor:
- código proveedor (E)
  - nombre proveedor (S)
  - fecha factura (E)
  - importe total (S) (se calcula)
    - \* ( código artículo (E)
    - precio unitario (S) (se saca del archivo Precios)
    - cantidad (E)
    - Importe (S)) (se calcula)
- Se ingresan todos los datos de entrada y se clikea sobre un botón Guardar, y se calcula y despliega el importe total
- Se cuentan el importe total y el de cada artículo como DET aunque sean campos derivados
  - La cantidad de DET depende de la cantidad de artículos que se ingresen
  - Tengo que agregar un DET por la posibilidad de una acción por el botón Guardar.
  - Se cuenta un solo DET por los datos de un artículo
-

25) En el mismo escenario anterior:

- a. Tengo que considerar una transacción adicional: la consulta implícita de ir a buscar el precio unitario
- b. **Se cuentan 2 RET**
- c. Se cuentan como DET solo los campos de entrada
- d. Todas las anteriores