1. Los productos de software personalizados están diseñados para satisfacer las necesidades generales del mercado. V/F

Los productos de software personalizados están diseñados para satisfacer las necesidades de un cliente en particular.

Los productos de software genéricos son los que están diseñados para satisfacer las necesidades generales del mercado

1. Los métodos discretos de recopilación de información no son suficientes por sí solos y deben complementarse con otros métodos. V/F

Los métodos discretos de recopilación de información son “menos perturbadores” que otras formas de averiguar información, por consecuencia, se consideran insuficientes cuando se utilizan por si solos, por lo que deben utilizarse junto a uno o más métodos.

1. La elicitación de requerimientos es una actividad principalmente técnica, con pocos desafíos sociales. V/F

La elicitación de requerimientos es el proceso de adquirir todo el conocimiento relevante para producir un modelo de los requerimientos de un dominio de problema.

Es una actividad principalmente de carácter social, más que tecnológico.

1. Las historias de usuario se utilizan principalmente en proyectos ágiles, pero son inapropiadas para proyectos con requisitos volátiles. V/F

Las historias de usuario son descripciones cortas y simples de un requerimiento de un sistema, escrito en lenguaje común y desde la perspectiva del usuario.

Las historias de usuario están hechas para proyectos de desarrollo agiles para la especificación de requerimientos, pero en cambio, los criterios de aceptación, una de sus ventajas es que están hechas para proyectos con requisitos volátiles o no muy claros.

1. El proceso de especificación de requerimientos dinámicos no toma en cuenta los cambios que ocurren a lo largo del tiempo en un sistema. V/F

El proceso de especificación de requerimientos dinámicos, como justamente lo dice la palabra, toma en cuenta los cambios que ocurren a lo largo del tiempo. Se considera que el sistema esta en un estado partículas hasta que un estimulo lo obliga a cambiar su estado.

Ejemplo de dinámicas: Redes de Petri, DTE, HU.

1. Un requerimiento funcional describe una restricción específica sobre cómo se debe construir el sistema, mientras que los no funcionales detallan qué debe hacer el sistema. V/F

Un requerimiento funcional describe lo que el sistema DEBE HACER, o incluso como NO DEBE comportarse.

En cambio, el requerimiento no funcional determina el COMO se debe haber algo. Pueden ser normas por cumplir, tiempos, uso de recursos, etc.

1. El modelo de diagramas de casos de uso facilita la participación activa de los usuarios en la definición de los requerimientos. V/F

Los casos de uso son el proceso de modelado de las funcionalidades del sistema en termino a eventos que interactúan entre los usuarios y el sistema.

Por esa razón, el uso de casos de uso alienta y facilita la participación activa de los usuarios.

1. Un caso de uso puede ser considerado un requerimiento, aunque no tenga un escenario asociado. V/F

Para que un caso de uso sea considerado un requerimiento debe estar acompañado de su respectivo escenario.

1. Las Redes de Petri son útiles para modelar sistemas concurrentes y garantizar que no ocurran condiciones de bloqueo. V/F

Las redes de Petri son utilizadas para especificar sistemas en tiempo real en los que son necesarios representar aspectos de concurrencia. Estos sistemas concurrentes al permitir tareas en simultaneo, deben estar sincronizadas para permitir la comunicación entre ellas y que no ocurran fallos como modificación de datos compartidos o condiciones de bloqueos.

1. El disparo de una transición en una Red de Petri puede generar un número ilimitado de tokens en los lugares de salida. V/F

El disparo de una transición en una red de Petri implica remover tokens de los lugares de entrada y distribuirlos en los lugares de salida, de forma limitada ya que depende de la cantidad de arcos que llegan y salen y los que llegan de la transición.

1. Para construir tablas de decisión, el analista necesita determinar el tamaño máximo de la tabla; eliminar cualquier situación imposible, inconsistencia o redundancia, y simplificar la tabla lo más que pueda. V/F

Las tablas de decisión para armarlas se debe identificar las condiciones y acciones, y luego, simplificar lo mas que se puede, sacando redundancia, condiciones opuestas y dejando solo 1, combinar tablas que sea evidente que una alternativa no representa una diferencia en el resultado.

1. Una tabla de decisión puede incluir condiciones no atómicas si son esenciales para el problema que se modela. V/F

Las tablas de decisión solo incluyen condiciones y acciones simples.

1. En la nivelación de un DFD (Diagrama de Flujo de Datos), las entradas y salidas de un proceso padre deben mantenerse en los diagramas hijos correspondientes. V/F

Forma

El contenido generado por IA puede ser incorrecto.

1. El modelo en cascada permite la iteración entre sus etapas, lo cual lo hace flexible para adaptarse a cambios en los requisitos. V/F

El modelo en cascada es lineal y secuencial, ósea que, cada etapa de desarrollo se debe completar antes que comience la siguiente.

1. El modelo en espiral combina actividades de desarrollo con la gestión de riesgos, permitiendo iteraciones y ajustes rápidos. V/F

El modelo en espiral combina actividades de desarrollo con la gestión de riesgos, incorpora objetivos de calidad, trata de mejorar los ciclo de vida clásicos y prototipos, y además, permite iteraciones, vuelta atrás y finalizaciones rápidas.

1. Un prototipo evolutivo se utiliza para construir rápidamente un sistema o parte de él con el objetivo de comprender mejor los requisitos. V/F

Un prototipo es un producto parcialmente desarrollado para que clientes y desarrolladores examinen algunos aspectos del sistema propuestos y deciden si es o no adecuado para el producto final.

El prototipo evolutivo construye rápidamente el sistema propuesto para comprender o aclarar aspectos y asegurar que el desarrollador, usuario y cliente tengan una compresión unificada de lo que se necesita.

1. El modelo de desarrollo por fases implica que dos sistemas funcionen en paralelo: uno en operación y otro en desarrollo. V/F

El modelo de desarrollo por fases se desarrolla de tal manera que se permite entregar por piezas. Esto implica que existen dos sistemas funcionando en paralelo: el operacional y el de desarrollo.

1. La norma ISO/IEC 25010 establece un marco para evaluar la capacidad y madurez de los procesos de una organización. V/F

El ISO/IEC 25010 define un modelo de calidad para productos de software, cubriendo características como funcionalidad, usabilidad, eficiencia, mantenibilidad, portabilidad, seguridad y compatibilidad.

En cambio, para evaluar la capacidad y madurez de los procesos de una organización, se utilizan normas como ISO/IEC 15504

1. En el ciclo de mejora continúa basado en ISO 9001, el modelo PDCA (Plan-Do-Check-Act) se utiliza para implementar y verificar mejoras en procesos de desarrollo. V/F

ISO 9001 o SGC (Sistema de gestión de calidad) o IRAM, implica un compromiso constante por identificar oportunidades de mejor, implementar cambios y evaluar su impacto.

El ciclo PDCA es un ciclo iterative para planificar, implementar, verificar y actuar sobre las mejoras.

1. En los sistemas de información, la calidad depende únicamente del software y no de factores como los datos o los procesos organizacionales. V/F

Un sistema de información abarca mas que el aspecto meramente computacional, ya que se debe tener en cuenta la forma de cómo se obtiene y organiza la información y el modo de organizar las herramientas.