INGE1 Promoción / FINAL

**Redes de Petri-Tablas de Decisión-DFD**

1. El DFD representa la transformación de entradas a salidas y es también llamado diagrama de burbujas. V/F

El DFD es el diagrama de flujo de datos, el cual permite visualizar un sistema como una red de procesos funcionales, conectados entre si por “conductos” y almacenamiento de datos.

Se le llama diagrama de burbujas.

Diagrama

El contenido generado por IA puede ser incorrecto.

1. Una tabla de decisión contradictoria tiene varias reglas que para las mismas condiciones especifican las mismas acciones. V/F

Una tabla de decisión puede ser 3 diferentes:

* Completa: Son aquellas que determinan acciones para todas las reglas y condiciones posibles.
* Redundante: Son aquellas que para reglas con mismas condiciones, tienen acciones iguales.
* Contradictoria: Son aquellas que para reglas con mismas condiciones, tienen diferentes acciones.

Así que, es falso ya que la que describe el enunciado es la REDUNDANTE.

1. El “Modelo de Comportamiento” del Análisis Estructurado define las interfaces entre el sistema y el ambiente donde el mismo se ejecuta. V/F

El modelo de comportamiento describe los cambios en el estado del sistema en respuesta a eventos, utilizando DTE (diagrama de transición de estados) o Redes de Petri.

La definición de interfaces entre el sistema y el ambiente donde el mismo se ejecuta, se representan en otros tipos de diagramas como DFD (flujo de datos).

1. Las tablas de decisión son utilizadas para especificar sistemas de tiempo real en los que es necesario representar aspectos de concurrencia. V/F

Las tablas de decisión permiten representar de forma concisa las reglas lógicas que hay que utilizar para decidir acciones a ejecutar en función de las condiciones.

Se usan para representar descripción de situaciones en forma clara y concisa que se resuelven por una decisión tomada en un momento específico del tiempo.

1. Una red de Petri se ejecuta disparando transiciones habilitadas. V/F

La red de Petri se ejecuta disparando las transiciones en las cuales el lugar de entrada tiene al menos tantos token como arcos hacia una transición.

**Modelos de Proceso**

1. El modelo en cascada requiere desarrolladores muy competentes. V/F

El modelo en cascada es un tipo de modelo tradicional. Además del modelo en cascada existe el modelo en V y el modelo basado en prototipos.

El modelo en cascada tiene una simplicidad que hace que sea fácil explicar a clientes, ya que es un modelo en el cual cada etapa de desarrollo debe ser completada antes que comience la siguiente.

Al ser simple, no requiere desarrolladores competentes.

1. La simplicidad del modelo en cascada hace que sea fácil explicarlo a los clientes. V/F

El modelo en cascada tiene una simplicidad que hace que sea fácil explicar a clientes, ya que es un modelo en el cual cada etapa de desarrollo debe ser completada antes que comience la siguiente.

1. El desarrollo por fases se caracteriza por combinar las actividades de desarrollo con la gestión del riesgo. V/F

El modelo que se caracteriza por combinar las actividades de desarrollo con la gestión del riesgo es el modelo en ESPIRAL.

El desarrollo por fases es el modelo en el que existen paralelamente la parte de desarrollo y la parte funcional, por lo cual, se puede entregar el sistema en piezas.

Diagrama

El contenido generado por IA puede ser incorrecto.

1. El modelo en espiral permite iteraciones, vuelta atrás y finalizaciones rápidas. V/F

El modelo en espiral se divide en 4 sectores: “Establecimiento de objetivos”, “valoración y reducción del riesgo”, “Desarrollo y validación” y “Planeación”.  
La ultima es en la cual se toma la decisión para la siguiente fase, por lo cual, el modelo si permite iteraciones, vuelta atrás y finalizaciones rápidas.

1. El modelo en espiral de Boehm trata de mejorar los ciclos de vida clásico y prototipo pero no incorpora objetivos de calidad y gestión de riesgos. V/F

El modelo en espiral mejora los ciclos de vida clásico y prototipo, y además, si incorpora objetivos de calidad y gestión de riesgos.

**Metodologías Ágiles (MA)**

1. Uno de los valores de las metodologías ágiles es tener documentaciones completas. V/F

Las metodologías agiles priorizan la entrega del software funcional por sobre la documentación extensa o completa.

Existen 4 valores de metodologías agiles:

- Fomenta la adaptabilidad en lugar de la estricta rigurosa planificación.

- Enfatiza la importancia de la comunicación personal y el trabajo en equipo mas que procesos y herramientas.

- Prioriza la entrega del software funcional por sobre la documentación completa.

- Se centra en asociarse con los clientes en lugar de negociar contratos formales.

1. En las MA los cambios de requerimientos son bienvenidos, aún tardíos, en el desarrollo. V/F

Uno de los principios de las metodologías agiles es que los cambios de requerimientos son bienvenidos, aun tardíos, en el desarrollo. Los procesos agiles capturan los cambios para que el cliente obtenga ventajas competitivas.

1. Según las MA los stakeholders, desarrolladores y usuarios no pueden ser capaces de mantener una comunicación constante. V/F

Los procesos agiles promueven un desarrollo sostenible, por lo tanto, si deberían de ser capaces de mantener una comunicación constante.

Además, los desarrolladores y usuarios deben trabajar juntos durante todo el proyecto.

1. El Product Backlog es la lista que contiene toda la funcionalidad que el equipo se comprometió a desarrollar durante un sprint determinado V/F

El “Product Backlog” es la lista que contiene toda la funcionalidad deseada del producto. Su característica más importante es que la funcionalidad se encuentra ordenada por prioridad.

En cambio, la “Sprint Backlog” es la lista que contiene toda la funcionalidad que el equipo se comprometió a desarrollar durante un sprint determinado.

Además, existe el “Burndown chart” que muestra un acumulativo del trabajo hecho día a día.

1. Una de las prácticas del proceso de XP es “Optimizar el todo” V/F

El “Extreme Programming” se basa en la simplicidad del código, desarrollo a pasos, pequeñas mejoras, corrección de código antes de añadir una nueva funcionalidad, etc. Pero la practica de optimizar el todo es del “SCRUM”.

**Calidad**

1. La calidad se puede definir como ausencia de defectos e imperfecciones V/F

La calidad es un termino totalmente subjetivo, que va a depender del juicio de la persona que intervenga en la evaluación.

Se puede definir como la propiedad o conjunto de propiedades inherentes a algo, que permiten juzgar su valor.

1. La calidad del software se divide en calidad del producto obtenido y del proceso de desarrollo y ambas son independientes. V/F

La calidad del software esta dividida en calidad del producto obtenido y el proceso de desarrollo, pero estas son dependientes una de la otra.

1. La familia ISO 9000 es un conjunto de normas de “gestión de calidad” aplicables solo a empresas de producción. V/F

La familia ISO 9000 es un conjunto de normas de “gestión de calidad” aplicables a cualquier tipo de organización, con el objetivo de obtener mejoras en la organización.

1. El CMMI posee una sola vista para su aplicación. V/F

El CMMI posee dos vistas que permiten un enfoque diferente según las necesidades de quien vaya a implementarlo.

Las dos vistas son:

- Escalonado: Centra su foco en la madurez de la organización. Tiene 5 niveles de madurez.

- Continuo: Enfoca las actividades de mejora y evaluación en capacidad de los diferentes procesos. Son 6 niveles de capacidad.

Gráfico

El contenido generado por IA puede ser incorrecto.

1. El objetivo de la ISO 90003 es proveer las especificaciones de como aplicar la ISO 9001 a los servicios. V/F

El objetivo de la ISO 90003 no es proporcionar especificaciones de cómo aplicar la ISO 9001 a los servicios, sino dar una guía sobre cómo aplicar la ISO 9001 específicamente en el desarrollo, suministro y mantenimiento de software. La ISO 90003 complementa la ISO 9001 al abordar las particularidades y requisitos de calidad en la industria del software.