Cuestionario

Instrucciones: Selecciona la respuesta correcta para cada pregunta.

1. ¿Qué es un requisito en el desarrollo de software?

a) Un conjunto de instrucciones para programadores.
b) Una condición o capacidad que el sistema debe cumplir.
c) Un documento técnico sobre seguridad informática.
d) Un diagrama de la estructura del código.

2. ¿Cuál de los siguientes es un ejemplo de un requisito funcional?

a) La aplicación debe cargarse en menos de 3 segundos.
b) El usuario puede registrar una cuenta con su correo electrónico.
c) El sistema debe soportar hasta 10,000 usuarios simultáneos.
d) La interfaz debe ser intuitiva y atractiva.

3. ¿Cuál de las siguientes opciones describe mejor un requisito no funcional?

a) El usuario debe poder iniciar sesión.
b) La aplicación debe ser accesible para personas con discapacidad.
c) El usuario podrá enviar mensajes a otros usuarios.
d) La aplicación debe permitir la edición de fotos.

4. ¿Qué representa una historia de usuario?

a) Un documento extenso con especificaciones detalladas.
b) Una breve descripción de una funcionalidad desde la perspectiva del usuario.
c) Un diagrama de flujo de la aplicación.
d) Un conjunto de pruebas unitarias.

5. ¿Qué son los criterios de aceptación en una historia de usuario?

a) La lista de errores detectados en la funcionalidad.
b) Las condiciones que deben cumplirse para que la historia de usuario sea considerada completa.
c) Las reglas de negocio que rigen toda la aplicación.
d) Las pruebas automatizadas del sistema.

6. ¿Qué es el backlog del producto en metodologías ágiles?

a) Un listado de errores pendientes por corregir en la aplicación.
b) Un informe final del estado del proyecto.

c) Una lista priorizada de requisitos y funcionalidades pendientes.
d) Un documento técnico sobre arquitectura de software.

7. En Scrum, ¿qué es un sprint?

a) Un período de trabajo donde se desarrolla un conjunto de funcionalidades.
b) Una reunión de planificación de requisitos.
c) Un método para evaluar el rendimiento del equipo.
d) Una lista de pruebas unitarias.

8. ¿Qué describe mejor la metodología Scrum?

a) Un marco de trabajo ágil basado en sprints y roles específicos.
b) Un conjunto de reglas estrictas de documentación.
c) Un enfoque de desarrollo en cascada tradicional.
d) Un modelo de prueba de software basado en prototipos.

9. ¿Cómo funciona Kanban en la gestión de proyectos ágiles?

a) Usa tableros y tarjetas para visualizar el flujo de trabajo.
b) Divide el proyecto en varias fases rígidas y consecutivas.
c) Utiliza solo documentos escritos para organizar tareas.
d) Prioriza la documentación sobre la ejecución del proyecto.

10. ¿Quién es el Product Owner en Scrum?

a) La persona que prioriza y define los requisitos en el backlog.
b) El líder técnico del equipo de desarrollo.
c) El responsable de escribir código para las funcionalidades.
d) El encargado de realizar pruebas automatizadas.

11. ¿Cuál es la función del Scrum Master?

a) Codificar nuevas funcionalidades.
b) Dirigir el equipo de marketing.
c) Facilitar el proceso Scrum y eliminar obstáculos del equipo.
d) Definir los requisitos del cliente.

12. ¿Cuál es el propósito de la reunión diaria (Daily Scrum)?

a) Planificar el presupuesto del proyecto.
b) Compartir avances, obstáculos y próximos pasos del equipo.
c) Redactar la documentación del sistema.
d) Diseñar la interfaz de usuario.

13. ¿En qué consiste el desarrollo incremental?

a) Se desarrolla todo el sistema de una vez y luego se lanza.
b) Se construye el software en pequeñas partes funcionales.
c) Se documentan los requisitos antes de comenzar a programar.
d) Se desarrollan varias versiones en paralelo.

14. ¿Cómo funciona el desarrollo iterativo?

a) Se repiten ciclos de desarrollo mejorando el producto progresivamente.
b) Se realiza una sola versión final del software.
c) Se desarrolla solo la parte más crítica del sistema.
d) Se eliminan funcionalidades no documentadas.

15. ¿Qué es la elicitación de requisitos?

a) El proceso de escribir código para el software.
b) La recopilación y análisis de necesidades de los interesados.
c) La generación de pruebas unitarias para validar requisitos.
d) La documentación final del sistema.

16. ¿Qué describe mejor la especificación de requisitos?

a) Un documento con los requisitos detallados del sistema.
b) Un conjunto de diagramas de flujo.
c) Un código de ejemplo de la aplicación.
d) Un informe de fallos del sistema.

17. ¿Qué implica el análisis de requisitos?

a) Evaluar y refinar los requisitos para que sean claros y factibles.
b) Implementar la base de datos del sistema.
c) Codificar los requisitos de usuario en una aplicación.
d) Diseñar la interfaz gráfica del usuario.

18. ¿Cuál es la diferencia entre validación y verificación de requisitos?

a) La validación asegura que los requisitos cumplen con las expectativas del cliente, mientras que la verificación comprueba que estén bien documentados y sin ambigüedades.
b) Son términos intercambiables que significan lo mismo.
c) La verificación se realiza antes de la validación en todos los casos.
d) La validación es solo para requisitos funcionales y la verificación solo para no funcionales.

19. ¿Qué es un caso de uso?

a) Una descripción de cómo un usuario interactúa con el sistema.
b) Un documento de requerimientos técnicos.

c) Un código de prueba para la aplicación.
d) Un diagrama de la estructura de la base de datos.

20. ¿Quiénes son los stakeholders en un proyecto de software?

a) Solo los programadores y testers.
b) Todas las personas o grupos interesados en el éxito del software.
c) Solo el cliente final.
d) Solo los gerentes de proyecto.
