

Trabajo Práctico – Metrics

Cuestionario

- 01- ¿Cuáles son las diferencias clave entre Metrics y Measure? Da ejemplos.
- 02- ¿Cómo podemos definir metrics?
- 03- Define qué es el Goal-Question-Metric y explica sus 3 etapas.
- 04- Explica las 3 categorías en las que se pueden clasificar las software metrics.
- 05- ¿En qué grupos podemos dividir el Process Quality Metrics? Nombra los elementos de cada grupo.
- 06- ¿Cómo podemos definir "test execution productivity"?
- 07- ¿Qué es el test coverage?
- 08- ¿Qué es el Defect Removal Efficiency (DRE) y cómo se calcula?
- 09- ¿En qué grupos podemos dividir el Product Quality Metrics? Nombra los elementos de cada grupo.
- 10- ¿Qué información relevante obtiene el test lead/manager con las métricas que le proporcionamos?

Respuestas

01. Metrics se refiere a la medición de un proceso o producto en términos cuantitativos, mientras que Measure se refiere a la medición de un atributo específico de un proceso o producto. Por ejemplo, una métrica podría ser el número total de defectos encontrados en una prueba, mientras que una medida podría ser el tiempo que tarda una prueba en ejecutarse.
02. Las métricas se pueden definir mediante el enfoque Goal-Question-Metric (GQM), que implica establecer objetivos de medición, formular preguntas específicas para alcanzar esos objetivos y luego seleccionar las métricas adecuadas para responder a esas preguntas.
03. El enfoque Goal-Question-Metric (GQM) consta de tres etapas: conceptual, operativa y cuantitativa. En la etapa conceptual, se establecen los objetivos de medición y se definen las preguntas específicas que se deben responder. En la etapa operativa, se identifican los elementos que se medirán y se seleccionan las métricas adecuadas para medirlos. En la etapa cuantitativa, se recopilan y analizan los datos para responder a las preguntas formuladas en la etapa conceptual.
04. Las software metrics se pueden clasificar en tres categorías: Process metrics, Product metrics y Project metrics. Las Process metrics miden la calidad del proceso, las Product metrics miden la calidad del producto y las Project metrics miden la calidad del proyecto.
05. Los Process Quality Metrics se pueden dividir en dos grupos: Metrics for Process Efficiency y Metrics for Process Effectiveness. Los elementos de Metrics for Process Efficiency incluyen tiempo de ciclo, tiempo de espera, tiempo de procesamiento y tiempo de entrega. Los elementos de Metrics for Process Effectiveness incluyen tasa de defectos, tasa de fallas, tasa de éxito y tasa de cobertura.
06. Test execution productivity se refiere a la cantidad de pruebas que se pueden ejecutar en un período de tiempo determinado. Se puede medir mediante la división del número total de pruebas ejecutadas por el tiempo total empleado en la ejecución de esas pruebas.
07. Test coverage se refiere a la medida en que un conjunto de pruebas cubre el código fuente o los requisitos del software. Se puede medir mediante la división del número de elementos cubiertos por el número total de elementos que deben ser cubiertos.
08. Defect Removal Efficiency (DRE) es una métrica que mide la eficacia del proceso de prueba en la eliminación de defectos. Se calcula dividiendo el número de defectos encontrados durante la prueba por el número total de defectos encontrados durante todo el ciclo de vida del software.
09. Los Product Quality Metrics se pueden dividir en dos grupos: Metrics for Software Product Health y Metrics for Defect Management. Los elementos de Metrics for Software Product Health incluyen tasa de defectos, tasa de fallas, tasa de éxito y tasa de cobertura. Los elementos de Metrics for Defect Management incluyen tasa de detección, tasa de corrección, tasa de reintroducción y tasa de resolución.
10. El Test Execution Report proporciona información relevante al test lead/manager sobre el progreso de las pruebas, incluyendo el número de pruebas ejecutadas, el número de pruebas exitosas y fallidas, el tiempo de ejecución de las pruebas y la cobertura de las pruebas. Esta información puede ayudar al test lead/manager a tomar decisiones informadas sobre el estado del proyecto y la calidad del software.