



ANÁLISIS Y DISEÑO DE SOFTWARE

BUENAS PRÁCTICAS EN LA CREACIÓN DE DIAGRAMAS UML

BUENAS PRÁCTICAS EN LA CREACIÓN DE DIAGRAMAS UML

A continuación, se detallan las principales buenas prácticas para crear diagramas UML de calidad:

1. Definir claramente el propósito del diagrama

Antes de comenzar cualquier diagrama, resulta esencial identificar el objetivo que se desea alcanzar. Cada tipo de diagrama UML tiene una función específica, por lo que entender para qué se está creando, evita incluir elementos innecesarios (Rodríguez, 2012).

Ejemplo. Si el propósito es mostrar la interacción entre objetos, se debe optar por un diagrama de secuencia, en lugar de intentar representarlo en un diagrama de clases.

2. Utilizar una notación estandarizada

El respeto por los símbolos, relaciones y convenciones propias de UML es fundamental. Modificar arbitrariamente los símbolos, puede llevar a malinterpretaciones.

Ejemplo. Una asociación entre clases debe representarse mediante una línea recta, no mediante flechas si no se trata de una relación de navegación.

3. Mantener la simplicidad y la claridad

Un diagrama recargado pierde su valor explicativo. Se recomienda incluir solo los elementos esenciales, utilizando agrupaciones o diagramas complementarios, si la complejidad crece demasiado.

Ejemplo. En lugar de modelar todas las clases de una aplicación en un solo diagrama, conviene dividirlo en varios diagramas, cada uno enfocado en un módulo o subsistema.

4. Nombrar correctamente los elementos

Los nombres de clases, casos de uso, actores, actividades y demás componentes deben ser claros, precisos y representativos de su función o naturaleza. Además, se sugiere usar la convención de nombrado correspondiente (por ejemplo, nombres de clases en singular y en mayúscula inicial).

Ejemplo. Un actor en un diagrama de casos de uso puede llamarse "Cliente" en vez de "Persona que compra productos".

5. Organizar el diagrama de manera lógica

El orden y disposición de los elementos, debe seguir un flujo natural de lectura, de preferencia de izquierda a derecha y de arriba hacia abajo. Esto ayuda a que quienes consultan el diagrama lo comprendan de forma rápida e intuitiva (Rodríguez, 2012).

Ejemplo. En un diagrama de actividad, las acciones deben seguir un orden descendente en correspondencia con la secuencia temporal de las actividades.

6. Indicar las relaciones, de manera precisa

Es importante detallar el tipo de relación entre los elementos, utilizando correctamente asociaciones, dependencias, generalizaciones, agregaciones o composiciones. Asimismo, se deben usar multiplicidades cuando corresponda, para indicar el número de instancias que intervienen en una relación.

Ejemplo. Entre una clase "Factura" y una clase "Producto", puede establecerse una relación de composición para indicar que una factura contiene productos específicos.

7. Revisar y validar los diagramas

Una vez terminado el diagrama, debe ser revisado cuidadosamente para detectar errores de concepto, inconsistencias gráficas o problemas de comprensión. La validación con otros miembros del equipo ayuda a asegurar que el modelo representa adecuadamente la realidad que se desea plasmar (Teniente López et al., 2015).

Ejemplo. Después de crear un diagrama de casos de uso, un analista puede reunirse con usuarios finales para asegurarse de que todos los casos reflejan las acciones reales que se esperan del sistema.

8. Actualizar los diagramas conforme evoluciona el sistema

El modelado UML debe acompañar la evolución del proyecto. Es fundamental que los diagramas se mantengan actualizados, para que continúen siendo útiles como referencia y guía (Rodríguez, 2012).

Ejemplo. Si se añade un nuevo rol al sistema, el diagrama de casos de uso debe ser actualizado para reflejar las nuevas interacciones.