

Contents

Clase 8: Verificación y Validación de Requisitos de Calidad - Parte 1: Conceptos Fundamentales y Técnicas de Verificación	1
---	---

Clase 8: Verificación y Validación de Requisitos de Calidad - Parte 1: Conceptos Fundamentales y Técnicas de Verificación

Objetivos de la Clase:

- Definir los conceptos de verificación y validación en el contexto de los requisitos de calidad.
- Comprender la diferencia entre verificación y validación.
- Conocer diversas técnicas de verificación de requisitos de calidad.
- Aplicar técnicas de verificación a ejemplos concretos de requisitos.

Contenido Teórico Detallado:

1. Introducción a la Verificación y Validación (V&V):

- **Definición de Verificación:** La verificación es el proceso de confirmar que el software se construye correctamente, es decir, que cumple con las especificaciones y requisitos definidos. En otras palabras, se enfoca en responder la pregunta: "¿Estamos construyendo el producto correctamente?". La verificación se realiza durante el desarrollo y se centra en evaluar artefactos intermedios, como documentos de diseño, código fuente y planes de prueba.
- **Definición de Validación:** La validación es el proceso de confirmar que el software construido satisface las necesidades y expectativas del usuario final y cumple con el propósito previsto. En esencia, responde a la pregunta: "¿Estamos construyendo el producto correcto?". La validación se realiza al final del desarrollo y se centra en evaluar el producto final en un entorno similar al real.
- **Diferencias Clave:** La diferencia fundamental radica en el enfoque. La verificación se centra en el *proceso* de desarrollo, asegurando que se sigan las especificaciones. La validación se centra en el *producto* final, asegurando que satisfaga las necesidades del usuario. La verificación es una actividad más interna y técnica, mientras que la validación implica la participación del cliente o usuario final.

2. Técnicas de Verificación de Requisitos de Calidad:

- **Revisiones Técnicas Formales (RTF):**
 - **Descripción:** Un proceso estructurado para evaluar los requisitos de calidad por un equipo de revisores (analistas, diseñadores, desarrolladores, testers). Involucra una preparación previa, una reunión formal de revisión y seguimiento de las acciones correctivas.
 - **Propósito:** Identificar ambigüedades, inconsistencias, omisiones, errores y falta de claridad en los requisitos de calidad.
 - **Proceso:**
 - (a) **Planificación:** Definir el alcance, el equipo de revisión y los roles.
 - (b) **Preparación:** Distribuir los documentos a los revisores para su análisis individual.
 - (c) **Reunión de Revisión:** Discutir los hallazgos y documentar los problemas identificados.
 - (d) **Retrabajo:** Corregir los requisitos basados en los problemas identificados.
 - (e) **Seguimiento:** Verificar que las correcciones se hayan implementado correctamente.
 - **Ejemplo:** Revisar los requisitos de rendimiento para asegurar que sean medibles y realistas, o revisar los requisitos de seguridad para asegurar que cubran todas las posibles vulnerabilidades.
- **Inspecciones:**
 - **Descripción:** Un tipo específico de RTF, aún más formal y riguroso, que se centra en la detección de defectos utilizando un proceso predefinido y listas de verificación (checklists).
 - **Propósito:** Detectar un alto número de defectos en los requisitos de calidad.

- **Características:** Uso de roles definidos (moderador, autor, revisores), listas de verificación detalladas, registro formal de defectos.
 - **Ejemplo:** Utilizar una lista de verificación para asegurar que todos los requisitos de seguridad incluyan consideraciones de autenticación, autorización, confidencialidad, integridad y disponibilidad.
- **Walkthroughs:**
 - **Descripción:** Una revisión informal por pares en la que el autor de los requisitos presenta el documento y otros participantes hacen preguntas y ofrecen sugerencias.
 - **Propósito:** Comprender los requisitos y detectar problemas superficiales.
 - **Características:** Menos formal que las RTF e inspecciones, enfoque en la comprensión y comunicación.
 - **Ejemplo:** El analista presenta los requisitos de usabilidad y el equipo de diseño ofrece feedback sobre su factibilidad y claridad.
 - **Análisis Estático:**
 - **Descripción:** Uso de herramientas automatizadas para analizar el documento de requisitos de calidad en busca de errores, inconsistencias, ambigüedades y posibles problemas.
 - **Propósito:** Identificar problemas que son difíciles de detectar manualmente.
 - **Ejemplo:** Utilizar una herramienta de análisis de texto para identificar frases ambiguas o redundantes en la documentación de requisitos.
 - **Prototipado:**
 - **Descripción:** Desarrollo de prototipos rápidos para validar la comprensión de los requisitos de calidad con los usuarios finales.
 - **Propósito:** Asegurar que los requisitos reflejen con precisión las necesidades y expectativas del usuario.
 - **Ejemplo:** Desarrollar un prototipo de interfaz de usuario para validar los requisitos de usabilidad y obtener feedback de los usuarios.
3. **Importancia de la Documentación:** Es crucial documentar todo el proceso de verificación, incluyendo los hallazgos, las acciones correctivas y el estado de los requisitos verificados. Esto permite la trazabilidad y facilita la gestión de cambios.

Ejemplos y Casos de Estudio:

1. **Caso de Estudio: Requisitos de Rendimiento para un Sistema de Comercio Electrónico:**
 - **Requisito Original:** "El sistema debe ser rápido." (Ambigüedad)
 - **Requisito Refinado:** "El tiempo de respuesta para la carga de cualquier página de producto debe ser inferior a 2 segundos en el 95% de los casos, con una carga máxima de 1000 usuarios concurrentes." (Específico y Medible)
 - **Verificación:** Se realiza una RTF para evaluar si el requisito refinado es factible desde el punto de vista técnico y si es consistente con otros requisitos del sistema. Se evalúa si se han considerado escenarios de carga pico y si se han definido métricas de rendimiento adecuadas.
2. **Ejemplo de Inspección:**
 - **Requisito de Seguridad:** "El sistema debe ser seguro." (Vago)
 - **Requisito Refinado:** "El sistema debe implementar autenticación de dos factores para todos los usuarios con privilegios administrativos, utilizando contraseñas con una longitud mínima de 12 caracteres, que incluyan letras mayúsculas, minúsculas, números y símbolos. Además, debe utilizar cifrado AES-256 para proteger la información sensible del usuario almacenada en la base de datos." (Específico y Detallado)
 - **Lista de Verificación (Checklist) para Inspección:**
 - ¿El requisito especifica el tipo de autenticación a utilizar? (Sí/No)
 - ¿Se define la longitud mínima y la complejidad de las contraseñas? (Sí/No)

- ¿Se especifica el algoritmo de cifrado a utilizar? (Sí/No)
- ¿Se identifican los datos sensibles a proteger? (Sí/No)
- ¿Se considera la gestión de claves de cifrado? (Sí/No)

Problemas Prácticos o Ejercicios con Soluciones:

1. Ejercicio 1:

- **Requisito:** "La aplicación móvil debe ser fácil de usar."
- **Tarea:** Identifica al menos tres problemas potenciales con este requisito desde el punto de vista de la verificación. Refina el requisito para que sea más verificable.
- **Solución:**
 - **Problemas:** Ambigüedad, falta de métricas, subjetividad.
 - **Requisito Refinado:** "El 80% de los usuarios deben completar la tarea de registro en la aplicación móvil en menos de 5 minutos, según lo medido en pruebas de usabilidad con 10 usuarios representativos del público objetivo."

2. Ejercicio 2:

- **Requisito:** "El sistema debe proteger la información del usuario."
- **Tarea:** Diseña una lista de verificación (checklist) para realizar una inspección de este requisito.
- **Solución:**
 - **Lista de Verificación:**
 - * ¿Se han identificado los tipos de información del usuario que deben protegerse? (Sí/No)
 - * ¿Se especifican los mecanismos de autenticación y autorización? (Sí/No)
 - * ¿Se utiliza cifrado para proteger la información sensible en tránsito y en reposo? (Sí/No)
 - * ¿Se implementan medidas de seguridad para prevenir ataques de inyección SQL y cross-site scripting (XSS)? (Sí/No)
 - * ¿Se realizan auditorías de seguridad periódicas? (Sí/No)
 - * ¿Se cumplen las regulaciones de privacidad aplicables (por ejemplo, GDPR, CCPA)? (Sí/No)

Materiales Complementarios Recomendados:

- **Libro:** "Software Engineering" de Ian Sommerville (Capítulo sobre Requisitos y Validación)
- **Artículo:** "IEEE Standard for Software Verification and Validation" (IEEE Std 1012)
- **Curso Online:** Cursos en plataformas como Coursera o edX sobre Ingeniería de Requisitos y Pruebas de Software.