Contents

Clase 5: Profundizando en los Requisitos de Calidad: Rendimiento y Seguridad

1. Objetivos de la Clase:

- Comprender en profundidad los requisitos de rendimiento y seguridad, su importancia y sus implicaciones en el diseño y desarrollo del software.
- Aprender a especificar requisitos de rendimiento cuantitativos y cualitativos.
- Identificar y clasificar diferentes tipos de amenazas a la seguridad del software.
- Comprender y aplicar principios básicos de diseño seguro.
- Aprender a documentar requisitos de rendimiento y seguridad de forma efectiva.

2. Contenido Teórico Detallado:

- **Requisitos de Rendimiento:**
  - **Definición:** Los requisitos de rendimiento especifican la velocidad, capacidad de respuesta, estabilidad y eficiencia con la que el software debe operar bajo diversas condiciones. Se refieren a la rapidez con que el software realiza sus funciones.
  - **Importancia:** Un rendimiento deficiente puede llevar a la frustración del usuario, abandono de la aplicación y pérdida de oportunidades de negocio.
  - **Tipos de Requisitos de Rendimiento:**
    - *Tiempo de Respuesta:* El tiempo que tarda el sistema en responder a una solicitud del usuario (e.g., tiempo de carga de una página web, tiempo de ejecución de una consulta a la base de datos).
    - *Troughput:* La cantidad de trabajo que el sistema puede procesar en un período de tiempo determinado (e.g., número de transacciones por segundo, número de usuarios concurrentes).
    - *Escalabilidad:* La capacidad del sistema para manejar un aumento en la carga de trabajo sin degradar significativamente el rendimiento (e.g., capacidad de añadir más servidores para manejar más usuarios).
    - *Utilización de Recursos:* La cantidad de recursos del sistema (CPU, memoria, disco) que se utilizan durante la operación.
    - *Latencia:* El tiempo de retraso en la transferencia de datos o en la ejecución de una operación.
  - **Especificación de Requisitos de Rendimiento:**
    - *Cuantitativos:* Expresados en términos numéricos medibles (e.g., "El tiempo de respuesta para la búsqueda de productos no debe exceder los 2 segundos"). Es crucial incluir unidades de medida y condiciones de prueba específicas (e.g., "bajo condiciones de carga pico de 1000 usuarios concurrentes").
    - *Cualitativos:* Expresados en términos descriptivos (e.g., "El sistema debe ser rápido y ágil"). Estos deben refinarse en requisitos cuantitativos siempre que sea posible. Por ejemplo, "El sistema debe proporcionar una experiencia de usuario fluida y sin retrasos notables al navegar por el catálogo de productos." Esto luego se puede traducir en tiempos de respuesta específicos.
- **Requisitos de Seguridad:**
  - **Definición:** Los requisitos de seguridad protegen el software y sus datos de accesos no autorizados, modificaciones, destrucción o divulgación.
  - **Importancia:** Las violaciones de seguridad pueden llevar a la pérdida de datos confidenciales, daños a la reputación de la empresa y problemas legales.
  - **Tipos de Amenazas a la Seguridad:**
    - *Malware:* Software malicioso diseñado para dañar o infiltrarse en un sistema (e.g., virus, gusanos, troyanos).
    - *Ataques de Denegación de Servicio (DoS/DDoS):* Ataques que inundan un sistema con tráfico para hacerlo inaccesible a los usuarios legítimos.
    - *Ataques de Inyección (SQL Injection, Cross-Site Scripting - XSS):* Ataques que explotan vulnerabilidades en el código para inyectar código malicioso y ejecutarlo en el sistema.

    - *Phishing:* Intentos de obtener información confidencial (e.g., contraseñas, números de tarjetas de crédito) haciéndose pasar por una entidad confiable.
    - *Ingeniería Social:* Manipulación de personas para obtener información confidencial o acceso a sistemas.
    - *Vulnerabilidades de Software:* Defectos en el código que pueden ser explotados por atacantes.
  - **Principios de Diseño Seguro:**
    - *Principio de Privilegio Mínimo:* Otorgar a los usuarios solo los privilegios necesarios para realizar sus tareas.
    - *Defensa en Profundidad:* Implementar múltiples capas de seguridad para proteger el sistema.
    - *Validación de Entrada:* Verificar que los datos ingresados por los usuarios sean válidos y seguros antes de procesarlos.
    - *Cifrado:* Utilizar el cifrado para proteger la confidencialidad de los datos en tránsito y en reposo.
    - *Autenticación y Autorización:* Verificar la identidad de los usuarios (autenticación) y controlar su acceso a los recursos (autorización).
    - *Gestión de Sesiones Segura:* Proteger las sesiones de usuario contra el secuestro.
    - *Registro y Monitorización:* Registrar eventos de seguridad y monitorizar el sistema para detectar anomalías.
  - **Documentación de Requisitos de Seguridad:** La documentación debe especificar los controles de seguridad implementados, los mecanismos de autenticación y autorización, y las políticas de gestión de contraseñas. Es importante identificar claramente los datos sensibles y cómo están protegidos.

3. Ejemplos/Casos de Estudio:

- **Caso de Estudio: Aplicación de Banca Móvil**
  - *Requisito de Rendimiento:* "La aplicación debe mostrar el saldo de la cuenta en menos de 1 segundo en el 95% de los casos, bajo una carga simulada de 5000 usuarios concurrentes realizando consultas de saldo."
  - *Requisito de Seguridad:* "La aplicación debe utilizar cifrado de extremo a extremo para proteger la información de la cuenta durante la transmisión." "Debe requerir autenticación multifactor (MFA) para el acceso a la cuenta."
  - *Amenaza:* Ataque Man-in-the-Middle que intercepte las comunicaciones entre la aplicación y el servidor.
  - *Mitigación:* Uso de HTTPS con certificados TLS actualizados y validación del certificado del servidor en la aplicación.
- **Caso de Estudio: Plataforma de Comercio Electrónico**
  - *Requisito de Rendimiento:* "La plataforma debe ser capaz de manejar al menos 1000 pedidos por minuto durante las horas pico sin experimentar una degradación significativa del rendimiento (tiempo de respuesta promedio no superior a 3 segundos)."
  - *Requisito de Seguridad:* "La plataforma debe proteger la información de las tarjetas de crédito de los clientes utilizando el estándar PCI DSS."
  - *Amenaza:* Ataque de inyección SQL en el formulario de inicio de sesión.
  - *Mitigación:* Uso de consultas parametrizadas y validación de entrada para prevenir la inyección de SQL.

4. Problemas Prácticos/Ejercicios con Soluciones:

- **Ejercicio 1:**
  - **Problema:** Define 3 requisitos de rendimiento (cuantitativos) para una aplicación de videoconferencia. Considera diferentes escenarios (e.g., reuniones con pocos participantes, reuniones con muchos participantes, compartir pantalla).
  - **Solución:**
    1. "El tiempo de latencia de audio y video no debe exceder los 200 ms para reuniones con hasta

10 participantes."
2. "La aplicación debe soportar al menos 50 participantes en una reunión sin una degradación significativa en la calidad del video (resolución mínima de 720p)."
3. "Compartir pantalla no debe consumir más del 30% de la CPU del usuario."

- **Ejercicio 2:**
  - **Problema:** Identifica 3 amenazas de seguridad comunes para una aplicación web y describe cómo mitigar cada una.
  - **Solución:**
    1. *Amenaza:* Cross-Site Scripting (XSS). *Mitigación:* Escapar y validar todas las entradas del usuario antes de mostrarlas en la página web. Utilizar una Content Security Policy (CSP) para restringir las fuentes de contenido que pueden ser cargadas por el navegador.
    2. *Amenaza:* Ataques de Fuerza Bruta en el inicio de sesión. *Mitigación:* Implementar un límite de intentos fallidos de inicio de sesión y bloquear la cuenta después de un cierto número de intentos. Utilizar contraseñas seguras y fomentar el uso de autenticación multifactor.
    3. *Amenaza:* Divulgación de información sensible en mensajes de error. *Mitigación:* Desactivar los mensajes de error detallados en el entorno de producción y registrar los errores para su análisis. Proporcionar mensajes de error genéricos al usuario.

5. Materiales Complementarios Recomendados:

- **Libros:**
  - "Writing Secure Code" by Michael Howard and David LeBlanc
  - "High Performance Web Sites" by Steve Souders
- **Artículos:**
  - OWASP (Open Web Application Security Project): Documentación y guías sobre seguridad web. (owasp.org)
  - NIST (National Institute of Standards and Technology): Publicaciones sobre seguridad informática. (nist.gov)
- **Cursos Online:**
  - Coursera: "Software Security" by University of Maryland
  - edX: "Cybersecurity Fundamentals" by Rochester Institute of Technology
- **Herramientas:**
  - OWASP ZAP (Zed Attack Proxy): Herramienta de pruebas de seguridad web.
  - JMeter: Herramienta para pruebas de rendimiento.