



Pruebas y mantenimiento de software

Lunes de 8:00 a 10:00
en CReCE

Prof. José Antonio Cervantes Alvarez
antonio.alvarez@academicos.udg.mx



Pruebas de sistema

- Una vez concluidas las pruebas de integración, el siguiente nivel de pruebas son las de sistema.
- Las pruebas de sistema verifican si el producto integrado satisface los requerimientos especificados.
- ¿Por qué son necesarias estas pruebas, si ya se aplicaron pruebas unitarias y de integración?
 - Las pruebas anteriores (unitarias e integración) son diseñadas y enfocadas a revisar las especificaciones técnicas del software.
 - **Las pruebas de sistema son diseñadas para evaluar el software desde la perspectiva del cliente y los futuros usuarios.**
 - El probador evalúa si los requerimientos fueron implementados de una forma apropiada.

- Muchas funciones y características de los sistemas son visibles y pueden ser probados después de integrar todos los componentes.
- Ejemplo:
 - El principal objetivo del sistema VSR es permitir que los clientes puedan comprar un carro de una manera fácil.
 - Para realizar la solicitud de compra, el usuario debe utilizar todos los componentes del sistema VSR: configurar el carro (DreamCar), calcular el financiamiento y la aseguradora (EasyFinance, NoRisk), enviar la orden a producción (JustInTime) y crear un contrato de compra-venta (ContracBase).
 - El sistema satisface su propósito sólo cuando todas las funciones del sistema y todos los componentes colaboran correctamente.



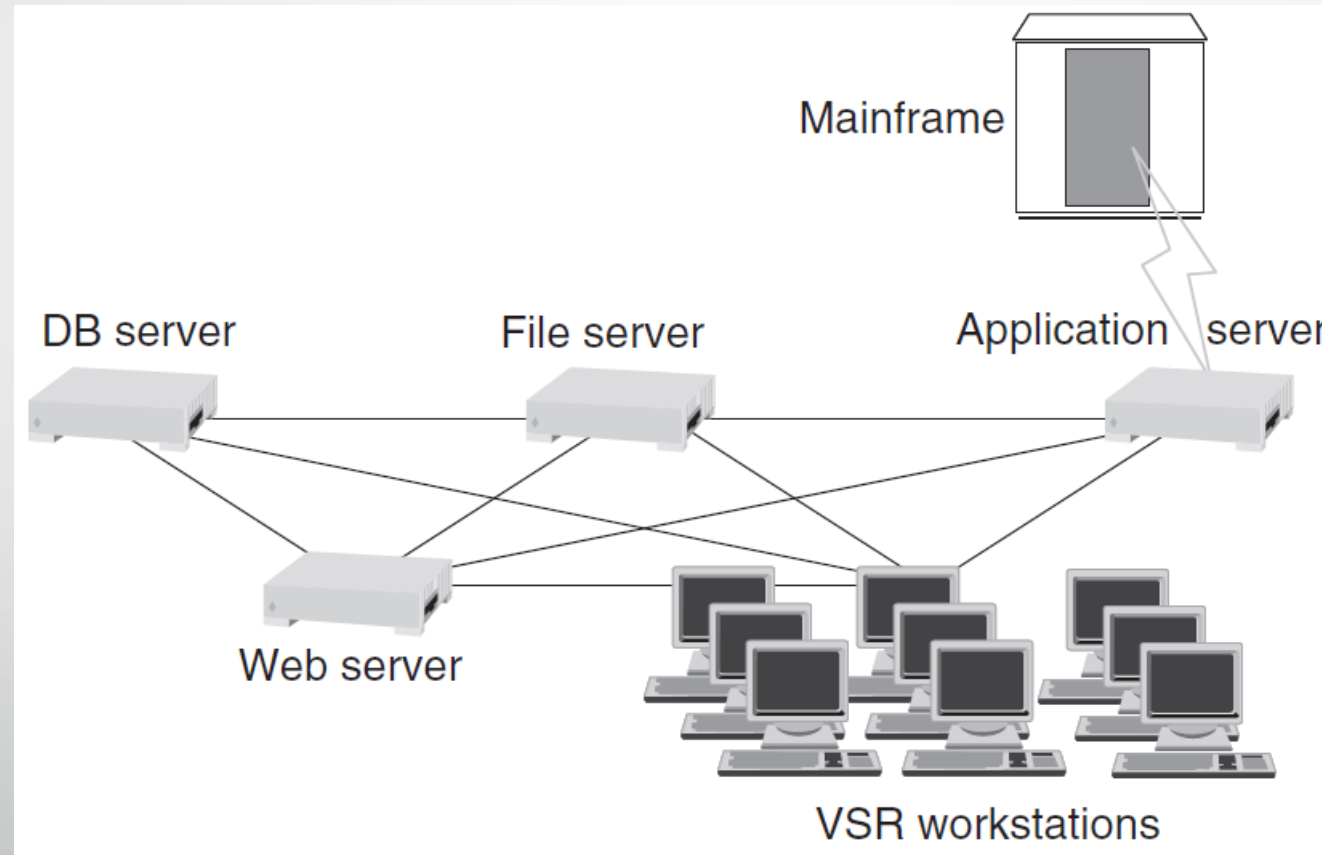
- Las bases para estas pruebas incluyen todos los documentos o información que describa los objetos de prueba a nivel de sistema.
 - Requerimientos del sistema.
 - Análisis de riesgos (si es que existen).
 - Manual de usuario.
 - Etc.



Objetos de prueba y el entorno de las pruebas

- Las pruebas de sistema prueban el sistema como un todo dentro de un entorno lo más similar posible al entorno de producción.
- El hardware y software que se utilizará en el entorno de producción debe ser instalado en la plataforma de pruebas.
 - Sistemas operativos.
 - Drivers.
 - Red.
 - Sistemas externos.
 - Etc.
- Las pruebas de sistema no solo se enfocan a probar el sistema, también se encargan de revisar el manual del sistema, el manual de usuario, el material de capacitación, etc.

- También la configuración y optimización del sistema deben ser cubiertas durante las pruebas de desempeño del sistema.



Ejemplo del entorno de pruebas del sistema VSR.



- Es importante considerar revisar la calidad de los datos en los sistemas que utilizan bases de datos o una gran cantidad de datos.
 - Los datos por si solos deben ser considerados objetos de prueba.
 - Se debe asegurar su:
 - **Atomicidad.** Cuando una operación consiste de una serie de paso, se debe garantizar que todos ellos se ejecutan o ninguno.
 - **Consistencia (integridad).** La Integridad de la Base de Datos nos permite asegurar que los datos son exactos y consistentes, es decir, que estén siempre intactos, sean siempre los esperados y que de ninguna manera cambien ni se deformen. De esta manera podemos garantizar que la información que se presenta al usuario será siempre la misma.
 - **Aislamiento.** Esta propiedad define cómo y cuándo los cambios producidos por una operación se hacen visibles para las demás operaciones concurrentes.
 - **Durabilidad.** Esta propiedad asegura que una vez realizada la operación, ésta persistirá y no se podrá deshacer aunque falle el sistema.
 - Otros aspectos a asegurar son: **seguridad** en el acceso a la información y **accesibilidad, disponibilidad**, entre otras.

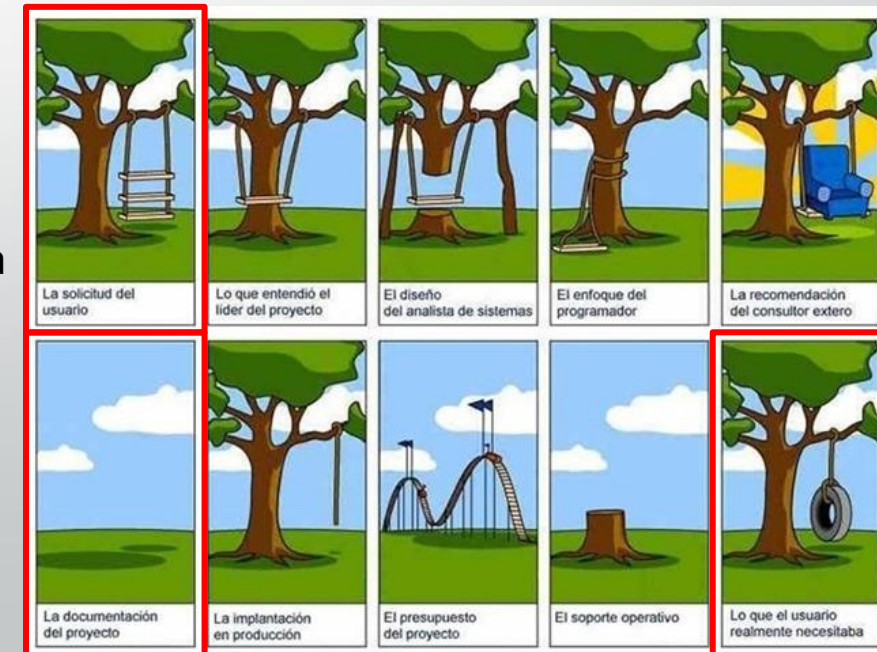
Objetivo de las pruebas

- El objetivo de las pruebas de sistema es validar que el sistema visto como un todo cumpla con los requerimientos funcionales y no funcionales que fueron especificados. Además de que éstos hayan sido implementados de forma completa y apropiada.
- Parte del objetivo de estas pruebas es detectar fallas generadas a partir de implementaciones incorrectas, incompletas o inconsistentes.
- Incluso requerimientos olvidados o no documentados deben ser identificados.



Problemas en las pruebas de sistema

- Es común que en la mayoría de los proyectos los requerimientos sean incompletos o no estén por escrito. Esto no permite que el probador sea capaz de comprender cuál debería de ser el comportamiento o funcionalidad esperada del sistema.
- Si no existen requerimientos todos los comportamientos del sistema pueden ser validos.
- La evaluación puede ser imposible de realizar.
- Sin embargo, los usuarios o clientes tienen una cierta perspectiva de lo que esperan que haga el sistema.





Pruebas de aceptación

- Antes de instalar y utilizar el software en un entorno real. Es recomendable desarrollar las pruebas de aceptación.
- Las pruebas de aceptación son las únicas en las cuales el cliente puede estar involucrado o que ellos pueden entender.
- Incluso, el cliente podría ser el responsable de estas pruebas.
- Las pruebas de aceptación pueden ser ejecutadas como parte de las pruebas de bajo nivel o distribuidas sobre todos los niveles de pruebas.



Pruebas de aceptación

- Existen cuatro formas típicas de las pruebas de aceptación:
 - **Pruebas de aceptación del contrato.** El cliente puede solicitar un conjunto de pruebas de aceptación como parte del contrato.
 - **Pruebas de aceptación del usuario.** Este tipo de pruebas son recomendadas cuando el cliente es diferente al usuario. Diferentes grupos de usuarios usualmente tienen diferentes expectativas de un nuevo sistema.
 - **Pruebas de aceptación operacional.** Asegura la aceptación del sistema a través de los administradores del sistema. Algunas de las pruebas pueden ser pruebas de respaldos, recuperación de desastres, administración de usuarios, revisión de las vulnerabilidades de seguridad.
 - **Pruebas de campo.** Cuando un software es diseñado para que opere en diferentes entornos o plataformas. Puede resultar muy caro o imposible realizar pruebas en todos los posibles entornos. Las pruebas de campo son una buena opción. Para ello, se libera una versión preliminar pero estable del sistema y se selecciona un número representativo de usuarios para probar el sistema.



Pruebas para nuevas versiones del producto

- La entrega del sistema solo marca el principio del ciclo de vida del software. Después de su instalación este será:
 - Usado por años o décadas,
 - modificado,
 - actualizado,
 - e incluso se agregarán nuevos módulos y funciones.
- Cada que sucede una modificación o actualización se crea una nueva versión del producto original.



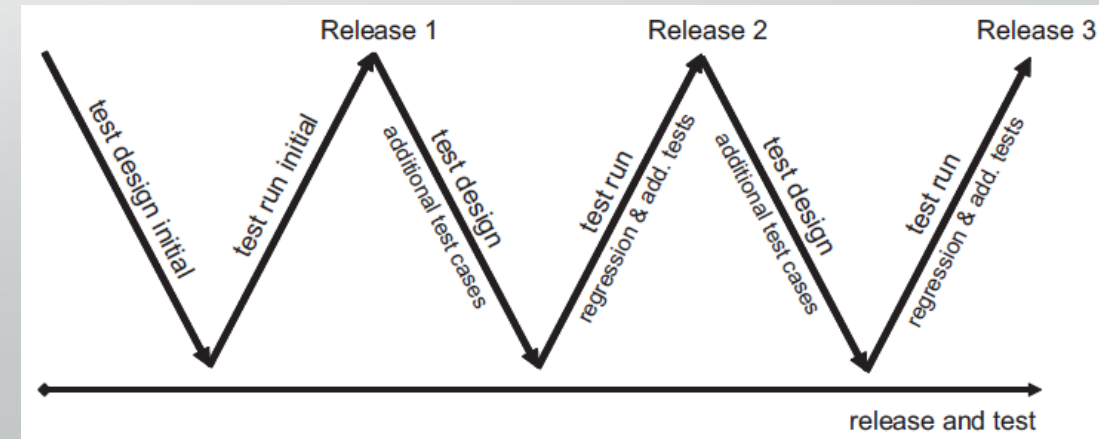
Mantenimiento del software

- **Mantenimiento preventivo.** Es un conjunto de medidas y acciones que se realizan para mejorar y optimizar el software, facilitando su óptimo rendimiento.
- **Mantenimiento adaptativo.** Este se da cuando el software es adaptado para funcionar bajo nuevas condiciones de operación.
- **Mantenimiento correctivo.** Se presenta cuando se descubren defectos del sistema después de la entrega e instalación del software.
- Las pruebas de cambio realizadas durante el mantenimiento (adaptativo o correctivo) pueden ser complejas debido a que:
 - Las especificaciones del sistema no están actualizadas o se desconocen.
 - Se trata de sistemas heredados (legacy systems).



Pruebas en desarrollos incrementales

- Un desarrollo incremental significa que el proyecto no es desarrollado como una sola pieza, sino que ha sido planificado para la entrega y liberación de pequeñas partes funcionales.
- El objetivo de este enfoque es garantizar que el sistema cumpla las expectativas y necesidades del cliente.
- Los prototipos son ejemplos de los modelos incrementales.
 - Rapid Application Development(RAD)
 - Rational Unified Process(RUP)
 - Evolutionary Development.
 - Extreme Programming(XP)
 - Dynamic Systems Development Method(DSDM)
 - SCRUM.





Tipos genéricos de pruebas

- **Pruebas funcionales.** Estas pruebas incluyen todos los tipos de pruebas utilizadas para verificar las entradas y salidas del sistema.
- **Pruebas no funcionales.** Los requerimientos no funcionales describen las características o atributos que debe tener el sistema tales como: confiabilidad, usabilidad, eficiencia, compatibilidad y seguridad. Algunas de las pruebas recomendadas son:
 - Pruebas de carga.
 - Pruebas de desempeño.
 - Pruebas de volumen.
 - Pruebas de estrés.
 - Pruebas de seguridad.
 - Pruebas de estabilidad o fiabilidad.
 - Pruebas de robustez.
 - Pruebas de compatibilidad y conversión de datos.
 - Pruebas de configuración del sistema.
 - Pruebas de usabilidad.
 - Revisión de la documentación.
 - Revisión de la mantenibilidad.



Tipos genéricos de pruebas

- **Pruebas de la estructura del software.** Para el desarrollo de estas pruebas se utiliza el conocimiento acerca del código estructurado de los objetos bajo prueba. Por lo general se prueba el flujo de control de los objetos. Estas pruebas son usadas principalmente en las pruebas unitarias y pruebas de integración.
- **Pruebas relacionadas a los cambios.** Cuando se implementan cambios, parte del código existente es modificado o se agregan nuevos módulos. Algunas modificaciones se dan como consecuencia de corregir fallos. Las pruebas relacionadas a los cambios debe verificar que los fallos encontrados ya no se presenten. También deben realizarse pruebas para verificar que no se hayan agregado nuevos defectos al sistema.



- Habrá un examen la próxima sesión.
 - **Temas a evaluar:** Todos los temas de la Unidad I y II.
 - **Duración:** 2 horas.