

Pruebas del software

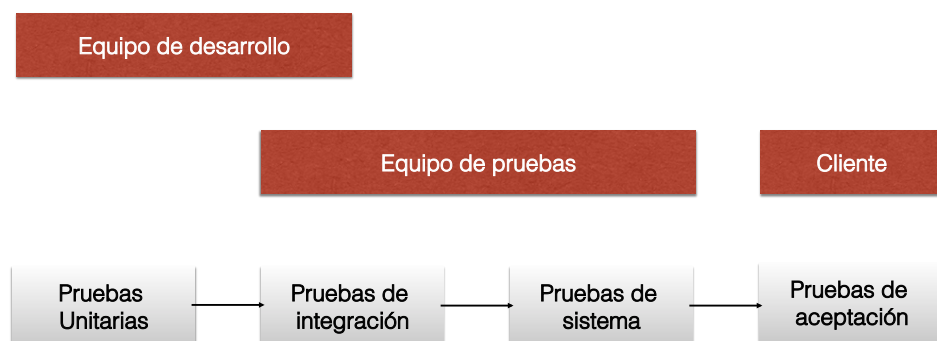
¿Qué es una prueba?

- “Conjunto de actividades planificadas y que se realizan de manera sistemática. Están orientadas a verificar y validar el software”
 - Verificación. El software implementa correctamente una función específica.
 - Validación. El software cumple con los requerimientos del cliente.

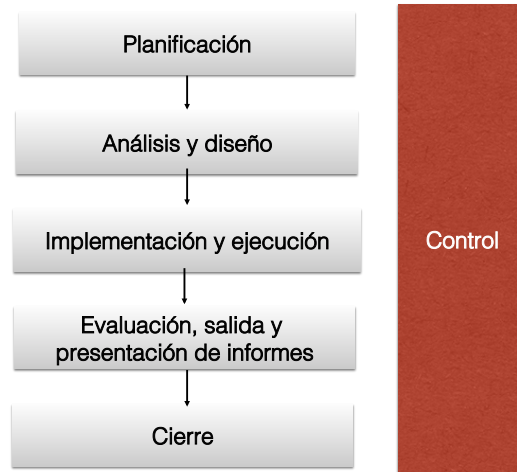
Características de las pruebas

- Destructivas. Diseñadas para buscar defectos y fallos.
- Ofrecen confianza sobre la calidad del software: los resultados obtenidos son los especificados y los esperados.
- Deben ser reproducibles.

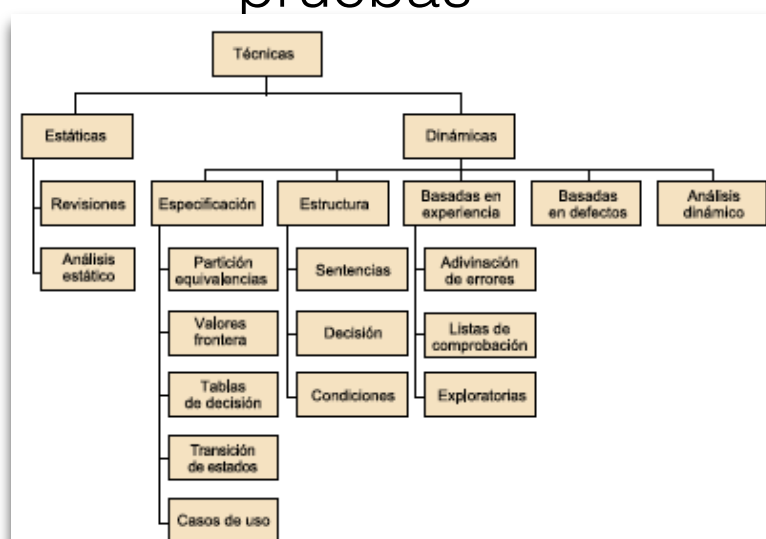
Niveles de las pruebas



Proceso de pruebas



Técnicas de diseño de pruebas



Técnicas de diseño de pruebas

- Especificación. Caja negra.
 - **QUÉ** hace el software.
- Estructura. Caja blanca.
 - **CÓMO** está implementado el software.
- Experiencia.
 - Basadas en la experiencia .
- Defectos.
 - Encontrar una o más categorías de defectos.
- Dinámico.
 - Análisis de la aplicación en ejecución.

Especificación

- Clases de equivalencia.
 - Dividir los casos en subconjuntos.
 - Rango de valores: $[i..j]$, $[-\infty..i]$, $[j..\infty]$
 - Lista: 1 categoría para cada valor
 - Validación: [Válido], [Inválido]
 - Booleano: [True], [False]
 - Otros: según el tipo de valores

Especificación

- Valores frontera
 - Es un refinamiento de la anterior.
 - Se añaden los valores frontera de las categorías.
 - Se incluyen los casos no válidos.

Inválida	Válida	Válida	Válida	Válida	Inválida
0	1	2..9	10..24	25..50	51

Especificación

- Tablas de decisión.
 - Cuando el sistema se comporta diferente según la combinación de valores de entrada.
- Basadas en estados
 - La salida depende del estado del sistema.
- Basadas en casos de usos
 - Si se han definido casos de usos, se pueden usar como base para hacer las pruebas.

Ejemplo

- CalculAreaTriangleRect (Valores frontera)
 - Juego de pruebas:
 - Rango. Según las especificaciones o el diseño:
 - p.e. [1,0..1.000.000,0]
 - Pruebas:
 - Selección de pruebas según categorías.
 - Evaluación: resultados teóricos y resultados reales.
 - Conclusiones de las pruebas.

Nota: Se muestra el juego básico, se podría completar con entradas no válidas: caracteres ('a'), símbolos ('\$'), expresiones (3+4),...

Prueba (valores)		catetb	catetc	Area		Perimetre	
Prueba de Fuego	cualq. válido	2	3	3.0	3.0	8.61	8.61
Inválida	$[-\infty..0.98]$	0.98 -2	0.98 -3	No Val	0.48 3.0	No Val	3.35 -1.39
Límite inferior	[0.99, 1.0, 1.01]	0.99 1.0 1.01	0.99 1.0 1.01	No Val 0.50 0.51	0.49 0.50 0.51	No Val 3.41 3.45	3.38 3.41 3.45
Válidas	[1.02..999999.98]	1.02 999999.98	1.02 999999.98	0.52 4999999979 52.00	0.52 499999997 952.00	3.48 3414213.50	3.48 3414213.50
Límite superior	[999999.99, 10^6 , 1000000.01]	999999.9 10^6 1000000.01	999999.99 10^6 1000000.01	4999999979 52.00 No Val	499999997 952.00	3414213.50 No Val	3414213.50
Inválida	[1000000.02 .. ∞]	1000000.02 10^{30}	1000000.02 10^{30}	No Val	499999997 952.00 inf	No Val	3414213.50 $3.4 \cdot E+30$

Ejemplo

- CalculAreaTriangleRect (Valores frontera)
 - Evaluación:
 - Funciona correctamente para los casos válidos.
 - En el límite superior e inferior da valores correctos, porque son valores válidos para la función.
 - En los valores no válidos inferiores da resultados que podrían parecer válidos pero que no tienen sentido.
 - En los valores no válidos superiores da un resultado de inf.
 - Conclusión:
 - Habría que modificar el código para detectar valores de entrada no válidos y dar un mensaje de error.