



2 • LAS PRUEBAS A TRAVÉS DEL CICLO DE DESARROLLO DE SOFTWARE

CAPITULO 2 • FUNDAMENTOS DE PRUEBAS DE SOFTWARE

BASADO EN EL PROGRAMA DE PROBADOR CERTIFICADO ISTQB
WWW.FULLADVANCED.COM

El ciclo de vida de desarrollo de software comprende las etapas por las cuales avanza un proyecto de software, describe cómo las actividades se relacionan entre sí, y en qué momento deberían llevarse a cabo.

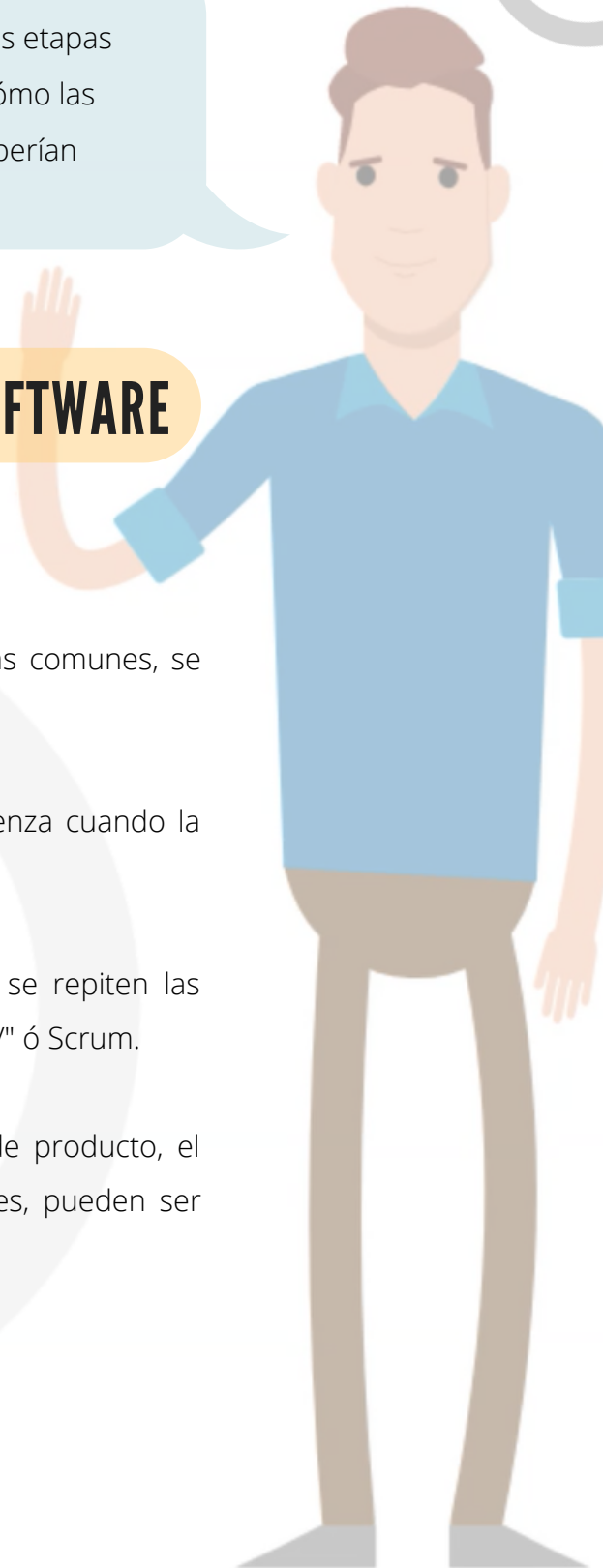
MODELOS DE CICLO DE DESARROLLO DE SOFTWARE

Los modelos de ciclo de vida de desarrollo de software más comunes, se clasifican en:

Modelos de desarrollo secuenciales: cada fase comienza cuando la anterior ha finalizado, como el modelo en cascada.

Modelos de desarrollo iterativos e incrementales: se repiten las fases tantas veces como sea necesario, como el modelo "V" ó Scrum.

Los modelos se seleccionan en base a las metas, el tipo de producto, el negocio, y los riesgos asociados, entre otras consideraciones, pueden ser combinados entre sí.



NIVELES DE PRUEBAS

Las pruebas de software son un conjunto de actividades organizadas y gestionadas en distintos niveles y relacionadas con su respectiva actividad de desarrollo:

- Pruebas de componentes
- Pruebas de Integración:
de componentes y sistema
- Pruebas de Sistema
- Pruebas de Aceptación:
 - Pruebas de Aceptación de usuarios,
 - Pruebas de Aceptación operativa,
 - Pruebas de Aceptación Contractual y de Regulación,
 - Pruebas Alfa y Beta



Los Stubs y Drivers nos permiten simular el comportamiento de las partes de software ausentes durante el desarrollo.

Stubs: Es llamado desde el componente que se está probando.

Driver: llama al componente bajo prueba.

COTS: son las siglas de Commercial off-the-shelf o software comercial de distribución masiva, por ejemplo Microsoft Office o Adobe Photoshop.



Buenas prácticas en la ejecución de las pruebas

- Cada ciclo de desarrollo debe contar con pruebas de validación y verificación
- Cada actividad de desarrollo tiene una actividad correspondiente de pruebas
- Cada nivel de pruebas tiene objetivos específicos
- El análisis y diseño de cada nivel de pruebas debe comenzar durante la actividad de desarrollo que corresponda
- Los probadores deben participar en las discusiones de definición y refinamiento de requerimientos y diseño

Nivel	Objetivos	Objetos de Prueba	Bases de Prueba	Defectos típicos	Responsables
Componentes	Conseguir defectos y fallas en objetos de prueba	Código / Estructuras de datos	Especificaciones de Componentes	Funcionamiento no acorde a la especificación	Realizado por el desarrollador
	Reducir el riesgo	Clases	Código	Problemas de flujos de datos	
	Construir confianza en la calidad del componente	Módulos de bases de datos	Modelo de datos	Código y lógica incorrecta	
	Prevenir defectos que escalen a niveles de prueba más altos				
Integración				Datos incorrectos, faltantes o mal codificados	Las pruebas de integración de <i>componentes</i> son a menudo responsabilidad de los desarrolladores Las pruebas de integración de <i>sistemas</i> son responsabilidad de los probadores
	Conseguir defectos y fallas entre las interfaces de módulos y sistemas, así como en los módulos en sí mismos	Subsistemas	Diseño de software y sistemas	Secuenciación o sincronización incorrecta a las llamadas de interfaz	
		Bases de Datos	Diagrama de secuencias	Incompatibilidad de la interfaz	
	Reducir el riesgo	Infraestructura	Especificaciones de interfaz y protocolos de comunicación	Fallos en la comunicación de componentes	
	Construir confianza en la calidad de las interfaces	Interfaces	Casos de uso	Fallos en la comunicación entre componentes manejados inadecuadamente o no manejados en absoluto	
	Prevenir defectos que escalen a niveles de prueba más altos	Interfaces de comunicación de aplicaciones API's	Arquitectura a nivel de componentes o sistemas		
Sistemas		Microservicios	Flujos de trabajo	Suposiciones incorrectas acerca del significado, unidades o límites de datos transmitidos entre componentes	Realizadas generalmente por probadores independientes
	Reducir el riesgo		Definiciones de interfaces externas		
	Verificar que los requerimientos del sistema son los diseñados y a su vez validarlos con lo especificado por el usuario	Aplicaciones	Especificaciones de Requisitos del sistema		
			Informes de análisis de riesgo	Cálculos incorrectos	
	Validar que el sistema está completo y funciona como se esperaba	Sistemas de hardware y software	Casos de Uso	Comportamiento incorrecto de tareas del sistema	
	Generar confianza en la calidad del sistema en su conjunto	Sistemas operativos	Épicas e historias de usuarios	Control de datos o flujos de datos incorrectos dentro del sistema	
	Encontrar defectos y fallas	Sistema bajo prueba	Modelos de comportamiento del sistema	Funcionamiento incorrecto del sistema	
	Evitar que los defectos escapen a niveles de prueba más altos, como las pruebas de aceptación o producción.	Configuración del sistema y datos de configuración	Diagramas de estado		
			Manuales de Sistema y de Usuario		

Nivel	Objetivos	Objetos de Prueba	Bases de Prueba	Defectos típicos	Responsabilidades
Aceptación			Generales: Procesos de negocio		
			Requisitos del usuario o del negocio		
			Regulaciones, contratos legales y estándares		
			Casos de uso		
			Requisitos del sistema		
			Documentación del sistema o del usuario		
	Establecer confianza en la calidad del sistema como un todo	Sistema bajo prueba	Procedimientos de instalación	Las reglas de negocio no se implementan correctamente	Clientes, usuarios de negocio, propietarios de productos o los operadores de un sistema y otras partes interesadas
	Validar que el sistema está completo y funciona como se espera	Configuración del sistema y datos de configuración	Informes de análisis de riesgos	El sistema no cumple los requisitos contractuales o reglamentarios	
	Verificar que los requerimientos funcionales y no funcionales del sistema cumplen con las especificaciones	Procesos empresariales para un sistema totalmente integrado	Operativas: Procedimientos de respaldo y restauración	Fallas no funcionales	
		Sistemas de recuperación y sitios activos	Procedimientos de recuperación ante desastres		
		Procesos operativos y de mantenimiento	Requisitos no funcionales		
		Formularios	Documentación de operaciones		
	Satisfacer requisitos legales o regulatorios.	Reportes Datos de producción	Instrucciones de despliegue e instalación		
			Objetivos de rendimiento		
			Paquetes de bases de datos		
			Normas o regulaciones de seguridad		



NIVELES DE PRUEBAS: Tipos de Pruebas de Aceptación

Pruebas de Aceptación de Usuario: se centra en validar que el sistema cumple los requisitos de funcionamiento esperado en un entorno operativo real o simulado.

Pruebas de Aceptación Operativa: valida si el sistema cumple con los requisitos de operación y la realizan los usuarios y los administradores de la aplicación.

Pruebas de Aceptación Contractual: se realiza según los criterios de aceptación de un contrato para producir software desarrollado a medida.

Pruebas Alfa y Beta: se realizan para recibir retroalimentación de usuarios, clientes y/u operadores potenciales o existentes antes de la liberación del producto de software al mercado.

Pruebas Alpha: se realizan en las instalaciones de la organización que desarrolla, pero no por el equipo de desarrollo.

Pruebas Beta: se realiza en las propias ubicaciones del cliente, fuera de la organización que desarrolla.



TIPOS DE PRUEBAS:

www.FullAdvanced.com por Julio César Oropeza

Pruebas Funcionales: se encargan del "qué" debe hacer el sistema.

Pruebas No Funcionales: prueban las características del producto de software, qué tan bien se ejecutan elementos como: recuperación, mantenibilidad, seguridad, resistencia a fallos, volumen, estrés y carga.

Pruebas de Caja Blanca: se enfocan en la estructura interna del sistema, entiéndase el código, arquitectura o flujos de datos dentro del sistema. Suele ejecutarse en el nivel de Componente e Integración.

Pruebas Relacionadas a Cambios:

Pruebas de Confirmación: garantiza que el defecto reparado pase las pruebas que previamente falló.

Pruebas de Regresión: garantizan que no se hayan generado nuevos problemas secundarios como consecuencia de los ajustes realizados al software.

Disparadores de Mantenimiento: Son modificaciones, migración y retiro del software.

Análisis de impacto evalúa las consecuencias de la implementación de un cambio en el software.



Click aquí para ver el capítulo completo

Pruebas de Mantenimiento: Suceden cuando se realizan cambios como parte del mantenimiento del software, evalúan el éxito del cambio en sí mismo y verifican posibles efectos secundarios. Es decir, constan de dos partes:

Probar los cambios y,
Pruebas de regresión.

