



# 1 • FUNDAMENTOS DE LAS PRUEBAS

## CAPITULO 1 • FUNDAMENTOS DE PRUEBAS DE SOFTWARE

V1.01 - ABRIL 2021

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

Existen **2 errores comunes sobre las pruebas** de software:

1. Sólo consisten en ejecutar el software y comprobar resultados.
2. Se enfocan en validar que el software cumpla con la verificación de requerimientos.



### Probar no es sólo correr pruebas.

#### Verificación:

¿Construimos bien el software según los requerimientos?

#### Validación:

¿Construimos el software adecuado basado en las necesidades reales del cliente?

## PROCESO DE PRUEBAS

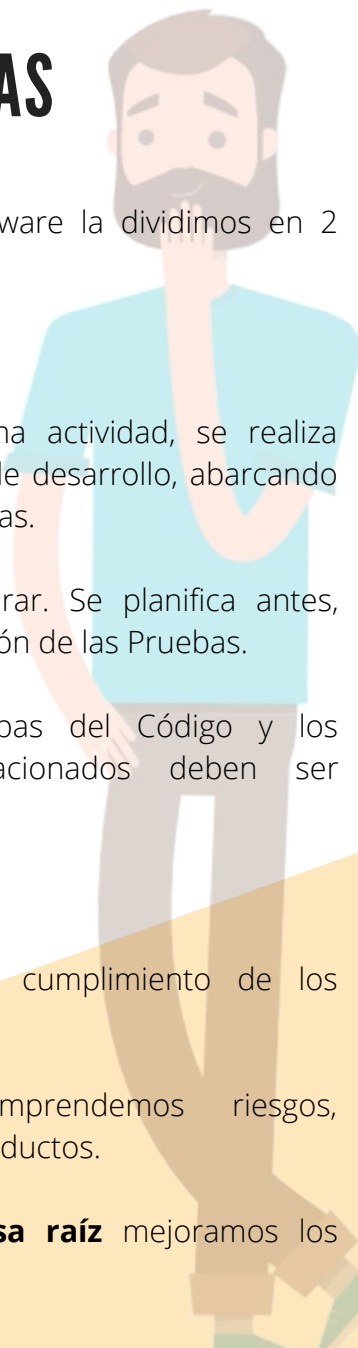
La definición de Pruebas de Software la dividimos en 2 partes: Proceso y Objetivos.

### El Proceso

- Probar es un proceso no una actividad, se realiza durante todo el Ciclo de vida de desarrollo, abarcando las pruebas Estáticas y Dinámicas.
- Las pruebas se deben preparar. Se planifica antes, durante y después de la ejecución de las Pruebas.
- Los resultados de las pruebas del Código y los Productos de Trabajo relacionados deben ser verificados.

### Objetivos

- Se **verifica** y se **valida** el cumplimiento de los requisitos.
- Al Detectar defectos comprendemos riesgos, mejoramos la calidad de los productos.
- Y cuando analizamos la **causa raíz** mejoramos los Procesos de Desarrollo



- Evaluar que todos los productos de trabajo estén correctos
- Verificar que todos los requerimientos especificados han sido satisfechos
- Validar que todos los requerimientos especificados son correctos, incluidos los de ámbito legal
- Generar confianza en el software
- Prevenir defectos
- Encontrar fallas y defectos
- Reducir el riesgo de fallas
- Proporcionar información para la toma de decisiones
- Reducir el nivel de riesgo de software de baja calidad
- Cumplir con estándares legales o comerciales



## Los objetivos de la prueba varían

dependiendo del sistema que se esté probando, en qué fase del ciclo de desarrollo se está, o del nivel de prueba.

---

## Probar es distinto de depurar

**Debugging o Depuración** es el proceso de desarrollo que encuentra, analiza y repara el defecto.

**Probar** es verificar que el software funciona como se espera.

---

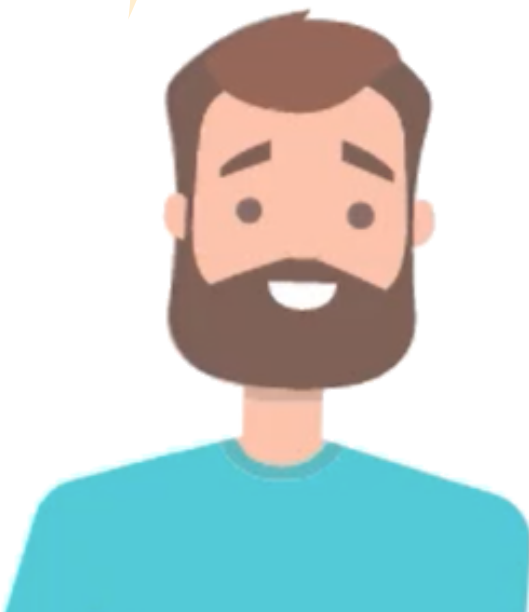
**El ciclo de depuración** implica que el desarrollador depura el código y realiza sus pruebas.

Luego del ciclo de depuración, un probador independiente ejecuta **las pruebas de confirmación**.

**Las pruebas de verificación** se realizan para garantizar que se cumpla lo establecido en los requerimientos.

## Producto de trabajo

Un producto de trabajo es cualquier documento, informe, reporte, diagrama e inclusive el código fuente o el sistema terminado.



# ¿POR QUÉ ES NECESARIO PROBAR EL SOFTWARE?

1. Todo el mundo comete errores, los errores introducen defectos que pueden generar fallas.
2. Reduce el riesgo y aumenta la confianza.
3. Algunos defectos son difíciles de identificar debido a que se originan en suposiciones erradas o puntos ciegos.
4. Algunas pruebas son obligatorias debido a estándares legales o comerciales.

## ¿Cómo contribuye la prueba al éxito de un proyecto?

1. Cuando los probadores y diseñadores trabajan en equipo se puede aumentar la comprensión del diseño del sistema y cómo probarlo.
2. Cuando los probadores y desarrolladores trabajan en equipo mientras el código se está desarrollando, le permite al probador ubicar segmentos de la aplicación susceptibles a error.
3. Tener probadores verificando y validando el software antes de su liberación aumenta la probabilidad de que el software cumpla con las expectativas del usuario.
4. Que el software cumpla con las expectativas del usuario aumenta la confianza en el producto, y la confianza en el producto genera fidelidad al proveedor.



## ***Aseguramiento de la Calidad y Pruebas son conceptos distintos***

**El Aseguramiento de la Calidad** se centra en el cumplimiento de los procesos adecuados para generar productos de trabajo de mayor calidad.

**El Control de la Calidad** implica varias actividades, incluidas las actividades de prueba, lo que apoya el logro de niveles apropiados de calidad.

**La Gestión de Calidad** une a ambos conceptos: Aseguramiento y el Control de la calidad.





¡Recuerda no confundir conceptos!

**Error:** Error, Mistake

**Defecto:** Falta, Defect, Bug, fault, fail.

**Falla:** Failure.



Un humano comete un **ERROR** que puede introducir un **DEFECTO** que podría generar una **FALLA**

## ¿ Por qué ocurren errores?

- Trabajo bajo presión
- Falibilidad humana
- Falta de experiencia del personal
- Complejidad del proyecto
- Falta de comunicación

## Causa Raíz de un defecto

Cuando se detecta una falla, debemos rastrearla para determinar la razón por la cual ocurrió. Los defectos deben ser analizados para identificar la raíz de su origen, y así reducir la posibilidad de que se repita en el futuro.

- No todos los defectos causarán fallas.
- No todas las fallas son de origen humano, pueden ser de origen ambiental.
- No todos los resultados inesperados de las pruebas son fallas.

Existen **falsos positivos** y **falsos negativos** en los resultados de las pruebas.

Si la prueba dice que hay un error cuando en realidad no lo hay, es un **Falso positivo**.

Si la prueba no detecta defectos que debió haber identificado, es un **Falso negativo**.

**Oráculo de prueba** es la fuente para determinar resultados esperados para compararlos con los resultados reales del software en pruebas.

# SIETE PRINCIPIOS DE LAS PRUEBAS

www.FullAdvanced.com por Julio César Oropeza



1. Las pruebas demuestran que hay errores, no su ausencia
2. Es imposible probarlo todo
3. Probar desde el inicio del ciclo ahorra tiempo, esfuerzo y dinero
4. Los defectos se agrupan
5. Los casos de prueba deben actualizarse para encontrar nuevos defectos
6. Las pruebas dependen del sistema que se esté probando
7. Es imposible que no hayan errores

La creencia de que el software debe ser liberado sin errores y con absoluta correctitud es algo imposible de lograr. Sin embargo se puede reducir el riesgo de falla en ambientes operativos haciendo pruebas con estándares altos de calidad.

## PROCESO DE PRUEBAS: Actividades

### Planificación y Control: ¿Cuales son los objetivos y cómo se alcanzan?

En la **planificación** se debe asegurar la comprensión de las expectativas del usuario, cuáles son sus metas, objetivos y los riesgos que se deberán considerar. Se establecen las bases de prueba y el plan de pruebas a seguir. El **control o monitoreo** verifica que las actividades se están llevando a cabo según el plan y se realiza durante todo el proceso de pruebas.

**Análisis:** Se analizan los requerimientos levantados a fin de verificar entre otras cosas, factibilidad de la prueba, tiempo y recursos necesarios. Se responde a la pregunta: **¿Que probar?**

**Diseño** de los casos de prueba de alto nivel. Un caso de prueba es un grupo de valores, precondiciones y resultados esperados. Se debe responder **¿Cómo probar?**

**Implementación:** Aquí se crean o se completan los productos de prueba necesarios para la ejecución de la prueba.

A menudo implementación y ejecución son etapas combinadas, en esta etapa se debe preguntar **¿Está todo listo para ejecutar las pruebas?**

**Ejecución:** Se ejecutan los casos de prueba, se registran y reportan los resultados de la prueba.

**Cierre o compleción de la prueba:** Finalmente, se libera el software cuando los criterios de aprobación han sido satisfechos y se ejecutan las actividades de cierre del proyecto.

# PROCESO DE PRUEBAS: Actividades

www.FullAdvanced.com por Julio César Oropeza



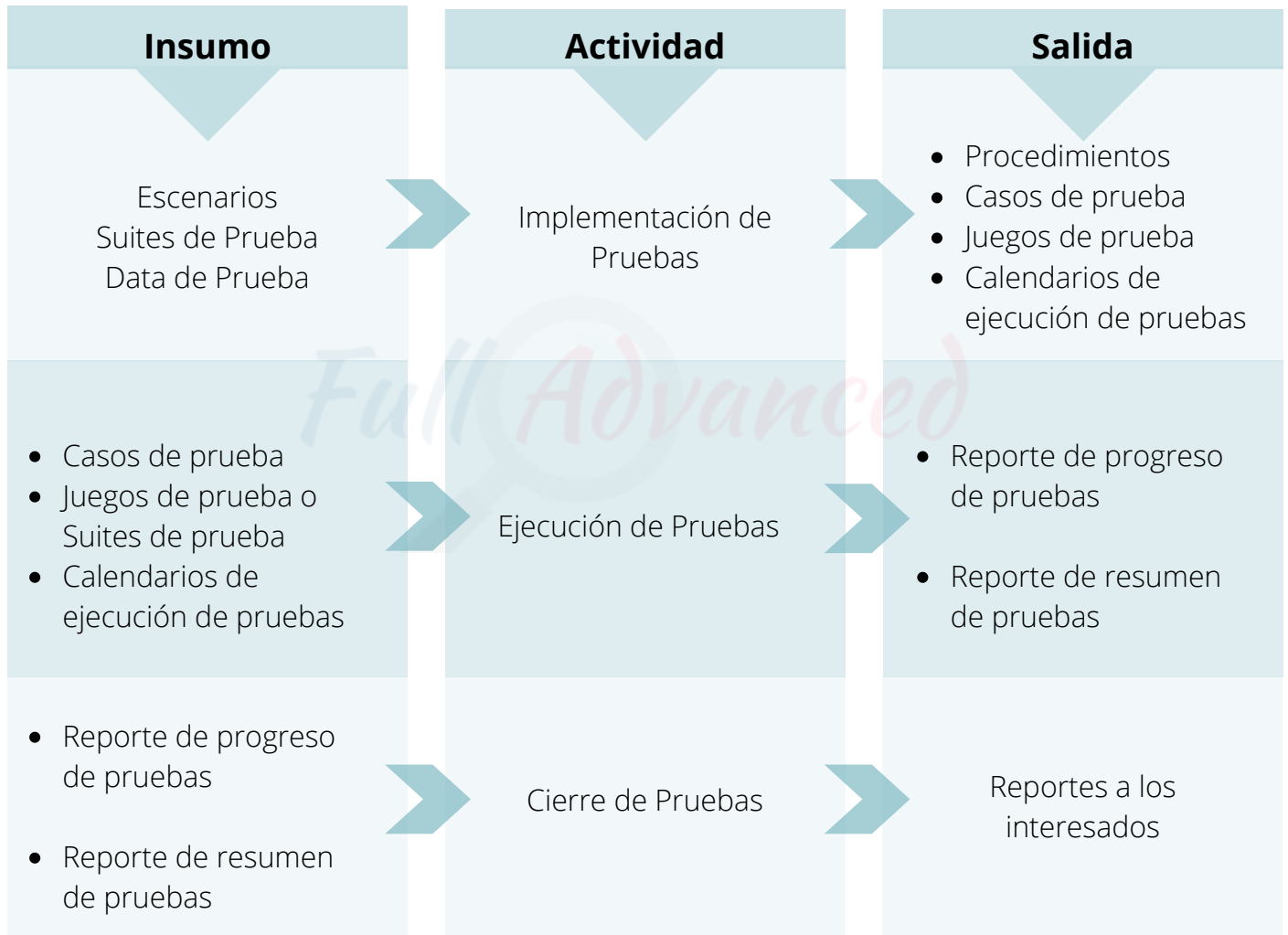
**Planificación → Análisis → Diseño → Construcción → Pruebas → Cierre**

Aunque parezca que el proceso de pruebas sigue una serie secuencial de actividades, algunas de ellas pueden llevarse a cabo iterativamente, como por ejemplo la implementación y la ejecución de las pruebas.

## PROCESO DE PRUEBAS: Productos de Trabajo

Insumo	Actividad	Salida
Bases de Prueba	Actividades de Planificación	Plan de Pruebas
Plan de Prueba	Control y Monitoreo	<ul style="list-style-type: none"><li>• Reporte de progreso de pruebas</li><li>• Reporte de resumen de pruebas</li></ul>
Bases de Prueba	Análisis de Pruebas	Condiciones de Prueba
Condiciones de Prueba	Diseño de Pruebas	<ul style="list-style-type: none"><li>• Escenarios</li><li>• Suites de Prueba</li><li>• Data de Prueba</li></ul>

# PROCESO DE PRUEBAS: Productos de Trabajo



## Trazabilidad entre Bases de Prueba y Productos de Trabajo

Trazabilidad entre Bases de Prueba y Productos de Trabajo					
Etapa					
Entradas					
Objetivos					
Actividades					
Salidas					
Trazabilidad					
Planificación de pruebas	Bases de Prueba	Definir los objetivos de las pruebas  Establecer la estrategia de pruebas: técnicas, tareas, fechas de entrega	Definir alcance, riesgos, enfoque y calendario de pruebas Establecer cómo medir el avance, quién hace qué y cómo lo hace Integrar las pruebas al ciclo de vida	Estrategia Plan de pruebas	A partir del plan de pruebas se genera trazabilidad hacia la totalidad de los productos de trabajo, ya que esta es la piedra angular del proyecto de pruebas.



# Trazabilidad entre Bases de Prueba y Productos de Trabajo

Trazabilidad entre Bases de Prueba y Productos de Trabajo					
Etapa	Entradas	Objetivos	Actividades	Salidas	Trazabilidad
Monitoreo y Control de pruebas	Cronograma de planificación	Verificar que el plan de pruebas cumpla con las fechas, estándares de calidad y demás especificaciones establecidas  Tomar acciones correctivas en caso de desvíos en la planificación	Revisar plan de pruebas vs progreso real	Reporte de progreso de pruebas  Reporte resumen de pruebas	Casos de pruebas Vs Calendario de pruebas
	Resultado de pruebas		Revisar resultados de pruebas vs. Criterios de salida (definición de listo)		Casos de pruebas Vs Riesgos identificados
	Reportes de análisis de riesgo		Determinar si más pruebas son necesarias		Casos de prueba vs Cobertura establecida
			Comunicar resultados a las partes interesadas		
Análisis de pruebas	Bases de Prueba	Evaluar las bases de prueba	Definir y priorizar las condiciones de prueba	Condiciones de prueba  Niveles de riesgo	Bases de prueba vs. Condiciones de prueba
Diseño de pruebas	Condiciones de Prueba	Diseñar el ambiente  Identificar data, infraestructura y herramientas necesarias	Definir y priorizar casos de prueba	Casos de prueba en alto nivel (Lógicos)	Condiciones de prueba vs. Casos de prueba de alto nivel
	Estrategia de prueba		Identificar data, herramientas e infraestructura		
	Plan de prueba		Diseñar el ambiente		
Implementación de pruebas	Casos de prueba	Desarrollar el ambiente de pruebas	Desarrollar y priorizar procedimientos de prueba   Test procedures	Casos de prueba de bajo nivel (concretos)	Procedimientos Escenarios Vs. Casos de prueba de bajo nivel
		Crear las suites de pruebas		Escenarios de prueba	
				Suites de prueba	
Ejecución de pruebas	Suites/Juegos de prueba	Ejecutar las pruebas	Comparar resultados obtenidos con los esperados: Oráculo de prueba	Reporte de resultado de pruebas	Defectos encontrados Vs. Cobertura establecida
				Reporte de defectos encontrados	Defectos encontrados Vs Riesgos identificados
Cierre de pruebas	Reporte de Resultados	Recopilar datos de actividades realizadas	Revisar que todos los defectos reportados hayan sido cerrados	Reporte a los interesados   Stakeholders	Pruebas realizadas Vs. Definición de hecho
			Almacenar los productos de trabajo que puedan ser reutilizables		



# PSICOLOGÍA HUMANA Y EL PROCESO DE PRUEBAS

El sesgo de confirmación es la dificultad para aceptar información que difiere con nuestras creencias actuales.

Es decir, es la tendencia a dar un mayor valor a las ideas u opiniones que confirman nuestras creencias.



## Sesgo de confirmación

Es común que algunas personas pueden percibir el proceso de prueba como una actividad destructiva, por esta razón probadores y jefes de prueba deben saber comunicarse de forma asertiva los resultados de las pruebas y construir relaciones positivas con sus colegas.



## Pruebas independientes

Los probadores independientes son personas distintas al autor del producto de trabajo.

Hay niveles de independencia, de menor a mayor:

- Pruebas realizadas por la persona que creó el producto bajo prueba
- Pruebas realizadas por otra persona del mismo equipo
- Pruebas realizadas por una persona de un equipo diferente
- Pruebas realizadas por una persona de una empresa diferente

## Los desarrolladores y probadores tienen esquemas de pensamiento diferente



**Desarrollador | Developer:** Se enfoca en diseñar y construir un producto.



**Probador | Tester:** Se enfoca en verificar y validar el producto, encontrando tantos defectos como sea posible antes de la liberación a producción.

## Maneras de comunicarse asertivamente entre equipos de desarrollo y Pruebas

- Recordar que es una colaboración, no una batalla entre bandos
- Enfatizar los beneficios de las pruebas
- Comunicar los defectos encontrados de una manera neutral, sin criticar al creador del producto de trabajo defectuoso
- Tratar de entender cómo se siente la otra persona
- Confirmar que el interlocutor ha entendido lo que se ha dicho, y viceversa

# VENTAJAS Y DESVENTAJAS DE TENER PRUEBAS INDEPENDIENTES

*Click para  
ver todo el  
material*



## Ventajas

Con probadores independientes aumenta la efectividad de las pruebas, lo cual es particularmente importante en grandes proyectos, o sistemas de seguridad crítica.

## Desventajas

El programador podría dejar toda la carga de las pruebas en el probador.

El probador puede aislarse del resto del equipo y tener problemas para alinearse a los objetivos del negocio.



Recuerda que el material completo lo puedes encontrar en **[www.FullAdvanced.com](http://www.FullAdvanced.com)**, en la sección de preguntas de cada capítulo puedes poner a prueba tu conocimiento con preguntas. Y si tienes dudas puedes dejar un comentario y te responderemos tan pronto como sea posible.