

# Capacitación Pruebas Unitarias (xUnit)

25 Noviembre 2022 – Sesión 8

**FUTURE  
AT HEART**

# Contenido



## Modulo I – Conceptos Básicos

- ¿Qué es una prueba unitaria?
- Otros niveles de prueba
- Conociendo las pruebas unitarias
- Técnicas de diseño de casos de prueba
- Principios FIRST

## Modulo II – Test Driven Development

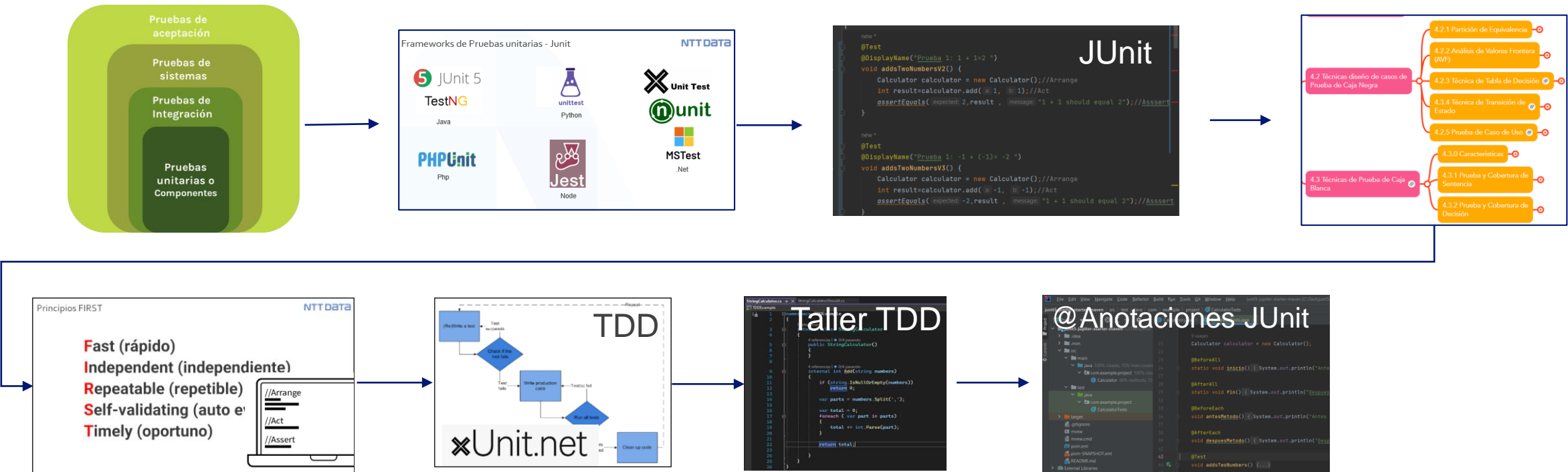
- Definición
- Desarrollo Ágil – Características
- Ciclo de desarrollo TDD
- Como escribir código que se pueda probar
- Cobertura de código
- Ventajas/Desventajas

## Modulo III – xUnit

- ¿Qué es xUnit?
- ¿Por qué xUnit?
- ¿Cómo funciona xUnit?
- Conceptos básicos
- Escenarios con xUnit
- Patrones de nombramiento de los escenarios
- Simulaciones
- Aserciones

## Modulo IV – Taller

- Preparación del Entorno
- Ejemplo práctico
- Creando casos de prueba
- Ejecución de casos
- Mutation Testing
- Conclusiones/Recomendaciones



## ¿Qué es la cobertura de pruebas unitarias?

La cobertura de pruebas unitarias es una métrica de QA que evalúa si los casos de prueba diseñados cubren el código de la aplicación y la cantidad de este código sometido a prueba cuando se ejecutan esos casos de prueba. Por lo tanto, la cobertura de pruebas ayuda a evaluar la efectividad de sus pruebas al ofrecer datos sobre varios elementos de cobertura.

### Resumen:

- La cobertura es la medida que visibiliza porcentaje del código esta siendo ejecutado por pruebas automáticas
- Esta representado por un %
- Que un proyecto tenga pruebas unitarias, no es garantía que estés probando bien





# Criterios para Calculo de Cobertura (1)

**1) Cobertura del producto:** Como su nombre lo indica, la cobertura del producto no es más que una cobertura de pruebas desde la perspectiva del producto. En otras palabras, es preguntarse sobre qué áreas del producto ha probado.

**Ejemplo:** Supongamos que se debe probar una aplicación simple como una calculadora. Aunque debe verificar las funciones esenciales como las cuatro operaciones aritméticas, eso no es suficiente. También debe considerar otros factores al probar la aplicación de la calculadora. Hay un esfuerzo adicional en las pruebas de varios escenarios, como lo bien que la calculadora maneja grandes cantidades. O, ¿qué pasa si el usuario hizo algo inusual como pegar caracteres especiales en el campo de texto?



**2) Cobertura de riesgos:** La cobertura de riesgos consiste en evaluar los riesgos involucrados en una aplicación y probarlos en detalle. Es necesario enumerar todos los posibles riesgos que pueden ocurrir en la aplicación y verificarlos adecuadamente.

**Ejemplo:** en una aplicación de entrega de alimentos, el usuario elige el restaurante y las preferencias de comida y paga a través de la pasarela de pago integrada. Un riesgo común es que los usuarios se desconecten cuando están en el proceso de pago. ¿Cómo se comportará la aplicación bajo este escenario? Asimismo, se deben considerar otros factores de riesgo relevantes que intervienen en la aplicación y comprobarlos.



**3) Cobertura de valor límite:** El análisis del valor límite es una parte de las pruebas de software que permite al *tester* crear los casos de prueba necesarios para un campo de entrada.

**Ejemplo:** un campo de entrada numérico solo debe permitir valores del 0 al 50. Por lo tanto, puede probar la aplicación ingresando números menores que 0, como negativos, y números mayores a 50; así como verificar que no se permitan escribir caracteres diferentes a números.

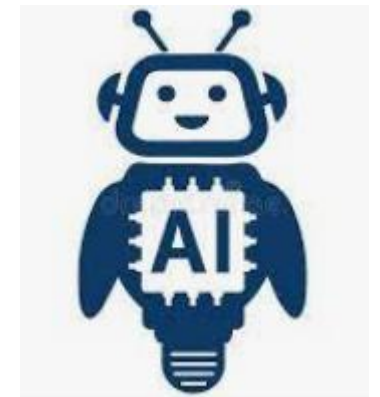


# Criterios para Calculo de Cobertura (2)

**4) Cobertura de requisitos:** La cobertura de los requisitos es la más crucial de todas las técnicas discutidas en este artículo. Por supuesto, incluso si su aplicación está libre de errores y funciona bien, ¿qué pasa si no cumple con los requisitos del usuario? Esto no solo pone en riesgo al proyecto, sino que afecta fuertemente la imagen de la empresa. Por eso el llamado a tener un plan de pruebas que valide que la aplicación cumpla con todos los requisitos. Después de todo, esa es toda la lógica detrás del desarrollo de software.



**5) Automatización de pruebas asistida por IA:** La automatización de pruebas asistida por IA (Inteligencia Artificial) es la técnica de cobertura de pruebas más avanzada de todas las mencionadas anteriormente. Esta utiliza herramientas de automatización que hacen que su enfoque de prueba pase al siguiente nivel. Estas herramientas asistidas por IA vienen con conjuntos de pruebas que aplican el aprendizaje automático (ML: *Machine Learning*) para aprender con cada ejecución. La IA incluso permite la autocorrección de casos de prueba, lo que alivia la carga del mantenimiento de la prueba; ayudando a obtener un sólido conjunto de pruebas que proporciona una cobertura de pruebas *premium*.



# ¿Por que es importante la cobertura?

- Encontrar y rectificar errores en etapas tempranas
- Crear más casos de prueba para garantizar una mejor cobertura
- Eliminar casos de prueba no deseados
- Posea un mejor control sobre su proyecto
- Ciclos de prueba eficientes
- Mayor ROI

## ¿Como calcular la cobertura de pruebas unitarias?

*Cobertura de pruebas = (Número de líneas que se han probado / Número total de líneas de código de la aplicación) x 100*

Si una aplicación que tiene 1000 líneas de código y 400 de ellas son ejecutadas por al menos un caso de prueba; su cobertura de pruebas es del 40%.

**GRACIAS**

**NTT DATA**

**FUTURE  
AT HEART**

