



Testing Efectivo en Java

Círculo Siete Capacitación

Clase 3 de 12
9 Agosto 2025

Ejercicio

- Laboratorio de TDD
 - Servicio de transferencias (MiniBank)
 - Carpeta ***labs***

Acceptance Test-Driven Development (ATDD)

- Define las pruebas de aceptación antes de escribir código.
- Involucra a negocio, QA y desarrollo para acordar qué significa que una funcionalidad esté “terminada”.
- Utiliza un lenguaje comprensible y estructurado para que todos entiendan los requisitos.
- Idea clave: ATDD asegura que construimos lo que el cliente realmente necesita, no lo que el desarrollador asumió.

En breve ATDD

- Negocio describe la necesidad en un lenguaje comprensible (generalmente Gherkin).
- QA transforma estas necesidades en pruebas de aceptación.
- Desarrollo implementa el código que haga pasar esas pruebas.
- ATDD se diferencia de TDD en que TDD parte de pruebas unitarias para guiar el diseño técnico, mientras que ATDD parte de pruebas de aceptación para guiar la funcionalidad completa.

Ciclo de ATDD

- Definir pruebas de aceptación
 - Se redactan los criterios de aceptación en lenguaje natural estructurado (ej. Gherkin).
- Automatizar las pruebas
 - Antes de escribir código, se implementan los step definitions que inicialmente fallarán.
- Desarrollar la funcionalidad
 - Escribir el código mínimo necesario para que las pruebas pasen.
- Refactorizar
 - Mejorar el código manteniendo las pruebas verdes.
- Este ciclo sigue el mismo espíritu Red → Green → Refactor de TDD, pero a nivel de comportamiento del sistema.

Beneficios clave

- Comunicación clara: todos entienden lo que significa "hecho".
- Prevención de malentendidos: reduce el riesgo de entregar algo que no es lo esperado.
- Pruebas vivas: los criterios de aceptación son ejecutables y siempre actualizados.
- Facilita BDD: ATDD es un puente natural hacia el Behavior-Driven Development.

Herramientas comunes en Java

- Cucumber → Automatiza escenarios escritos en Gherkin.
- Serenity BDD → Integra ATDD/BDD con reportes detallados.
- JUnit 5 → Base de ejecución de las pruebas.
- Selenium / RestAssured → Interacción con UI o APIs.

Comparativa TDD y ATDD

Característica	TDD	ATDD
Enfoque inicial	Pruebas unitarias	Pruebas de aceptación
Nivel de prueba	Bajo (código)	Alto (funcionalidad)
Participantes	Desarrollo	Desarrollo, QA, Negocio
Lenguaje	Código	Natural Estructurado

Tarea

- Terminar de implementar los tests pendientes de la práctica de TDD.

