**AUTOMATIZACION DE PRUEBAS E INTREGRACION CONTINUA**

ATOMATIZAION DE PRUEBAS: permite la ejecuciond e peubras si la inervencion manual, reduciendo tiempop de prueba y mejorando detecicon de errores.

Es un proceso mediant le cual se utilizan hermmainet srcipts para ejecutar pruebas autimaticas.

Su objetivo es eliminar la necesidad de pruebas manuales repetitivas y aumentas la confiabilidad en los resultados.

BENEFICIO:

• EFICIENCIA Y RAPIDEZ : se ejectuan en menos tiempo

• DETECCION TEMPRANA DE ERRORES:

• COBERTURA DE PRUEBAS MEJORADAS.

• CONSISTENCIA: reduce la variabilidade los resultados de pruebas

• REDUCCION DE COSTOS:inversonmás alt,a peor se reduce alargo plazo

### **Integración Continua (CI – Continuous Integration)**

**Definición:**  
La **Integración Continua** es una práctica de desarrollo en la que los desarrolladores **integran frecuentemente sus cambios de código en un repositorio compartido**, preferentemente varias veces al día. Cada integración se valida automáticamente mediante **builds (compilaciones) y pruebas automatizadas** para detectar errores lo antes posible.

**Objetivo:**  
➡️ Detectar errores rápidamente.  
➡️ Asegurar que el software siempre esté en un estado funcional.  
➡️ Reducir los conflictos de integración entre desarrolladores.

**Ejemplo:**  
Cada vez que un desarrollador sube (hace push) su código, se ejecutan automáticamente pruebas unitarias y una compilación del sistema para verificar que no haya errores ni regresiones.

### 🚀 **Entrega Continua (CD – Continuous Delivery)**

**Definición:**  
La **Entrega Continua** es una práctica que extiende la Integración Continua. Busca que el software esté **siempre en un estado desplegable (listo para producción)**. Luego de cada integración, el sistema pasa por pruebas más exhaustivas y se empaqueta automáticamente para ser entregado a un entorno de preproducción o producción con un simple clic o paso manual.

**Objetivo:**  
➡️ Minimizar el tiempo entre el desarrollo y la entrega al usuario.  
➡️ Asegurar lanzamientos frecuentes, confiables y predecibles.

**Ejemplo:**  
Después de que una nueva funcionalidad pasa las pruebas automáticas, el sistema está listo para ser desplegado en producción con un solo paso (por ejemplo, validación manual del Product Owner).

### 📌 Resumen comparativo

| **Concepto** | **Integración Continua (CI)** | **Entrega Continua (CD)** |
| --- | --- | --- |
| ¿Qué automatiza? | Pruebas y compilación tras cada cambio | Preparación y validación para desplegar |
| ¿Qué asegura? | Que el código integrado no rompa el sistema | Que el software esté siempre listo para producción |
| ¿Despliega a producción? | ❌ No, solo prueba | ✅ Puede sí (manual o automática) |
| ¿Cuándo se usa? | Durante el desarrollo | Luego de CI, antes de lanzar |