

# Prácticas de Testing Ágil

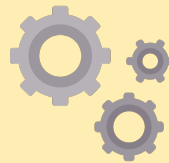
## Introducción

El testing de software tiene como objetivo encontrar defectos. Existen diferentes tipos de pruebas a las que se puede someter un software para confirmar que se lo está construyendo correctamente, es decir, libre de defectos.

### Pruebas de unidad e integración automatizadas

Las **pruebas unitarias** consiste en probar de forma individual las funciones. Debido a lo específicas que son, generalmente son las pruebas automatizadas de menor costo.

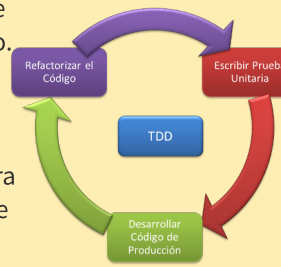
Las **pruebas de integración** verifican que los diferentes módulos y/o servicios usados por el software funcionen correctamente cuando trabajan en conjunto.



### TDD (Test Driven Development)

**TDD** es un estilo de programación que entrelaza codificación, testing y diseño.

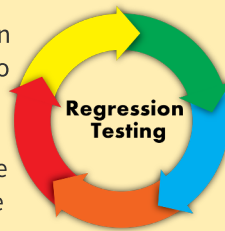
Para llevarlo a cabo se debe escribir primero las pruebas unitarias que describa un aspecto del software. Luego, se codifica sólo lo necesario para que las pruebas pasen. Y finalmente se mejora el código (refactoring)



### Pruebas de regresión a nivel de sistema automatizadas

Implica volver a probar el software completo después de haber corregido algún defecto, con el objetivo de asegurarse de no haber introducido uno nuevo.

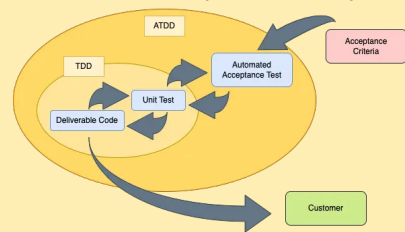
Son altamente automatizables ya que se tiene un conjunto de pruebas definidas que deben ejecutarse periódicamente luego de una corrección.



### ATDD (Desarrollo conducido por Pruebas de aceptación)

**ATDD** involucra al cliente, desarrolladores y testers, que colaboran para escribir pruebas de aceptación antes de

implementar la funcionalidad de software correspondiente. Estas pruebas de aceptación actúan como requerimientos para describir cómo debería funcionar el sistema.

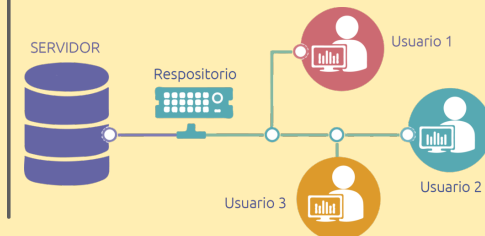


### Pruebas exploratorias

Se utilizan ampliamente en modelos ágiles e implican descubrimiento, investigación y aprendizaje. Enfatiza la libertad personal y la responsabilidad del evaluador individual.



### Control de version de las pruebas con el código



Siempre debe ser capaz de identificar las versiones de los scripts de prueba que van con una versión particular de código.

## Conclusion:

Agile testing ofrece un muy buen enfoque de pruebas y una gran cantidad de herramientas que nos ayudan a entregar un producto de calidad, el tener un enfoque en donde todo el equipo, incluido el cliente, está involucrado durante todo el proceso y que además existe una mentalidad enfocada a la calidad con un aire de respeto y mejora continua definitivamente ayuda a que podamos entregar un producto de calidad en tiempo record.

Grupo 5 - Ditta, Marcos - Ferraro, Nicolas - Mansilla, Julián

## Referencias:

- [www.agilealliance.org](http://www.agilealliance.org)
- Agile Testing: A Practical Guide For Testers And Agile Team - Lisa Crispin / Janet Gregory