

# Trabajo de Investigación: Pruebas de regresión

Chavarria Peña Jonathan Andrés  
Estudiante Ing. en Sistemas de Computación  
Universidad Fidélitas  
San José, Costa Rica  
[jonach1998@gmail.com](mailto:jonach1998@gmail.com)

Phillips Tencio Edmond  
Estudiante Ing. en Sistemas de Computación  
Universidad Fidélitas  
Alajuela, Costa Rica  
[ephillips10986@ufide.ac](mailto:ephillips10986@ufide.ac)

Morales Cordero Valeria  
Estudiante Ing. en Sistemas de Computación  
Universidad Fidélitas  
San José, Costa Rica  
[valemc0603@gmail.com](mailto:valemc0603@gmail.com)

Sánchez Camacho Carlos Daniel  
Estudiante Ing. en Sistemas de Computación  
Universidad Fidélitas  
San José, Costa Rica  
[csanchez20965@ufide.ac](mailto:csanchez20965@ufide.ac)

## Resumen—

### 1. INTRODUCCIÓN 2. DESARROLLO

#### 2.1. *Qué es una Prueba de Regresión*

Son pruebas para determinar si las aplicaciones existentes aún pueden funcionar como se espera después de que se hayan actualizado o modificado. Cada cambio pueden interrumpir la funcionalidad del software.

No se deben cambiar las pruebas de regresión hasta que las pruebas actuales pasen. Una fail en una prueba de regresión significa que una nueva funcionalidad ha afectado a otra que era correcta en el pasado.

Como también una falla en el test podría indicar que se ha vuelto a producir un error que ya había sido resuelto en el pasado.

#### 2.2. *Cuándo debe realizarse la prueba de regresión*

Es importante ejecutar este tipo de pruebas cada vez que cambia su código. Los cambios de aplicaciones que requieren pruebas de regresión son:

1. Mejoras.
2. Parches.
3. Cambios de configuración.
4. Integración con otro software.

Es necesario asegurarse de que las funciones del código antiguo puedan seguir funcionando normalmente después de que se incluya el nuevo código.

#### 2.3. *¿Cómo hacer una prueba de regresión?*

Se puede realizar de distintas formas, depende de los cambios realizados y de que tanta confiabilidad se necesita en los datos de la prueba realizada.

- Realizar pruebas a todo el código: Es la forma de garantizar que todos los casos de prueba en el programa sean realizados para su comprobar su integridad y funcionamiento. Este metodo suele ser costoso ya que a menudo requiere una gran inversión de tiempo y recursos.
- Realizar las pruebas en una parte específica del código: Permite seleccionar una parte específica de las pruebas que se ejecutaban, por lo tanto, no es tan costosa.
- Priorización de casos de prueba: Prioriza los casos de prueba de acuerdo con su impacto normalmente se prueban aspectos críticos y funcionalidades de uso frecuente.

#### 2.4. *Software a utilizar*

IntelliJ IDEA: Es desarrollado por JetBrains, es un IDE para JAVA.

#### 2.5. *Herramienta para las pruebas*

2.5.1. *JUnit*: Es un entorno de pruebas para Java creado por Erich Gamma y Kent Beck. Se encuentra basado en SUnit creado originalmente para realizar pruebas unitarias para el lenguaje Smalltalk.

2.5.2. *Unit tests*: Este tipo de tests consiste en probar de forma individual las funciones o métodos. Generalmente son pruebas automatizadas de menor costo, y pueden ejecutarse rápidamente por un servidor de integración continua.

## REFERENCIAS

- [1] ALEXYNIOR (2019). *¿Qué es una prueba de regresión?*. <https://adictec.com/que-son-pruebas-de-regresion/>
- [2] THE QA TESTING CHANNEL. (2018). *Pruebas de Regresión en 1 Minuto*. <https://www.youtube.com/watch?v=OEcIXnU6EQw>