

Testowanie oprogramowania

Przemysław Nowak

2019/2020

1 Spis treści

- Testowanie oprogramowania - Definicja
- Podział testowania
 - Testy jednostkowe(modułowe)
 - Testy integracyjne jednostkowe oraz systemowe
 - Testy systemowe
 - Testy akceptacyjne
- Przykładowy test jednostkowy
- Metoda TDD
- Bibliografia

Testowanie oprogramowania -

proces związany z wytwarzaniem oprogramowania. Jest to jeden z procesów zapewnienia jakości oprogramowania. Testowanie ma na celu weryfikację oraz walidację oprogramowania. Weryfikacja oprogramowania pozwala skontrolować, czy wytwarzane oprogramowanie jest zgodne ze specyfikacją. Walidacja sprawdza, czy oprogramowanie jest zgodne z oczekiwaniami użytkownika. Testowanie oprogramowania może być wdrożone w dowolnym momencie wytwarzania oprogramowania (w zależności od stosowanej metody).

- Testy jednostkowe(modułowe)
- Testy integracji modułów
- Testy systemowe
- Testy integracji systemów
- Testy akceptacyjne

Test jednostkowy -

metoda testowania tworzonego oprogramowania poprzez wykonywanie testów weryfikujących poprawność działania pojedynczych elementów (jednostek) programu – np. metod lub obiektów w programowaniu obiektowym lub procedur w programowaniu proceduralnym. Testowany fragment programu poddawany jest testowi, który wykonuje go i porównuje wynik (np. zwrócone wartości, stan obiektu, zgłoszone wyjątki) z oczekiwanymi wynikami – tak pozytywnymi, jak i negatywnymi (niepowodzenie działania kodu w określonych sytuacjach również może podlegać testowaniu).

Zaletą testów jednostkowych jest możliwość wykonywania na bieżąco w pełni zautomatyzowanych testów na modyfikowanych elementach programu, co umożliwia często wychwycenie błędu natychmiast po jego pojawieniu się i szybką jego lokalizację zanim dojdzie do wprowadzenia błędnego fragmentu do programu.

Testy integracyjne – sprawdzają poprawność integracji między pojedynczymi modułami lub komponentami oprogramowania. W przeciwieństwie do poziomych testów jednostkowych, na tym etapie nie sprawdzamy poprawności działania pojedynczych komponentów lub modułów. Przykładowym testem integracyjnym jest na przykład weryfikacja poprawności połączenia z bazą danych.

Testy systemowe są przeprowadzane na całym systemie i mają za zadanie sprawdzić konkretny scenariusz z punktu widzenia użytkownika.

Przykładowym scenariuszem, może być na przykład dodanie artykułu przez system CMS. Od testera wymaga to więc przeprowadzenia autoryzacji, uruchomienia edytora, napisania artykułu, publikacji i weryfikacji czy wszystko jest poprawnie wyświetlane na stronie. Są to jedne z najtrudniejszych testów, bowiem trudno je zautomatyzować.

Testy akceptacyjne to ostatni poziom testów, są one wykonywane w celu akceptacji produktu przez klienta. Generalnie w tym momencie nie powinno już być żadnych błędów w oprogramowaniu. Testy te są przeprowadzane na środowisku produkcyjnym i powinny sprawdzać czy oprogramowanie ostatecznie realizuje wszystkie założenia i oczekiwania klienta.


```
class ExampleTest extends PHPUnit_Framework_TestCase {  
  
    public function testGreetings()  
    {  
        $greetings = 'Hello_World';  
        $this->assertEquals( 'Hello_World', $greetings )  
    }  
}
```

```
PHPUnit 4.3.4 by Sebastian Bergmann.  
Configuration read from F:\Localhost\unittests\phpunit.xml  
.  
Time: 417 ms, Memory: 2.25Mb  
OK (1 test, 1 assertion)
```

Rysunek: Efekt testu

TDD (Test-Driven Development) jest techniką tworzenia oprogramowania (nie jest to technika pisania testów), w której główną ideą jest w pierwszej kolejności pisanie testów do nieistniejącej funkcjonalności, a dopiero potem napisanie kodu implementującego tę funkcjonalność.

Cykl:

- 1 Napisanie przypadku testowego
- 2 Uruchomienie napisanego testu
- 3 Napisanie minimalnego kodu do przejścia testu
- 4 Refaktoryzacja kodu
- 5 Powtarzamy cykl



https://pl.wikipedia.org/wiki/Testowanie_oprogramowania



https://pl.wikipedia.org/wiki/Test_jednostkowy



[http:
//strefakodera.pl/programowanie/testowanie-oprogramowania/
testowanie-oprogramowania-pozioomy-testow](http://strefakodera.pl/programowanie/testowanie-oprogramowania/testowanie-oprogramowania-pozioomy-testow)



[https://blog.onwelo.pl/
czym-jest-technika-tdd-i-jak-wyglada-jej-cykl-zycia/](https://blog.onwelo.pl/czym-jest-technika-tdd-i-jak-wyglada-jej-cykl-zycia/)

Koniec

Dziękuję za uwagę