

Mity na temat testowania oprogramowania

2010-02-05 13:53:39

Przetestowanie całej aplikacji/systemu jest możliwe. Jeśli testy zostaną odpowiednio zaplanowane to możliwe jest znalezienie i poprawienie wszystkich błędów w aplikacji.

Oprogramowanie zawiera coraz więcej linii kodu, staje się coraz bardziej złożone. Coraz więcej aplikacji współpracuje ze sobą w ramach jednego, rozproszonego systemu. Pokrycie testami wszystkich ścieżek przejścia we wszystkich kombinacjach jest niemożliwe choćby z powodu ograniczeń związanych z budżetem i czasem.

Kiedy projekt zakłada, że po ostatniej fazie testów aplikacja będzie wolna od błędów, dział testów staje się odpowiedzialny za każdy problem znaleziony w wersji produkcyjnej. Z kolei próba przetestowania wszystkich aspektów tworzonej aplikacji spowoduje opóźnienie w dostarczeniu wersji produkcyjnej. W rzeczywistości prawie każda aplikacja zawiera błędy. Pytanie, jakiego są one rodzaju i jak często objawiają się w np. niestabilnym działaniu gotowego produktu.

Testy powinny być przeprowadzane w pełni skonfigurowanym środowisku.

Im bardziej środowisko testowe przypomina docelowe środowisko produkcyjne, tym bardziej można polegać na wynikach testów. Jeśli środowisko produkcyjne klienta jest przez niego w pełni kontrolowane, testowanie może odbywać się w jak najbardziej zbliżonych warunkach. Jeśli tak nie jest, testy przeprowadzone w pełni skonfigurowanym środowisku mogą spowodować, że ważne przypadki testowe mogą w ogóle nie powstać.

Testowanie jest potrzebne, aby udowodnić, że oprogramowanie nie zawiera błędów.

Testowanie samo w sobie służy do poprawienia jakości aplikacji, ale jest tej jakości miernikiem. Przystępując do testów nie należy robić tego z założeniem, że system działa.

Zazwyczaj przypadki testowe, których wykonanie nie spowodowało wykrycia błędu oznacza się statusami "passed", "ok", "tested ok". Jednak z punktu widzenia testera to właśnie Test Case, który pozwolił na odkrycie błędu jest sukcesem.

Oprogramowanie oprócz tego, że powinno poprawnie wykonywać pożądane i zaimplementowane funkcjonalności, nie powinno pozwalać na operacje, do których wykonywania nie zostało stworzone.

Automatyzacja testów. Jeśli można coś zautomatyzować należy to zrobić.

Automatyzacja jest potrzebna. Oszczędza nam, testerom długich wieczornych godzin spędzonych przy manualnych testach regresyjnych. Z kolei estymowana liczba godzin na testy manualne, może skłonić projekt do zainwestowania w automatyzację testów.

Jednak co z czasem spędzonym na nauce narzędzi do testów automatycznych, tworzeniu skryptów i ich aktualizacją? Jedynie odpowiednia kombinacja testów automatycznych, testów eksploracyjnych i manualnych testów wykonywanych na podstawie przypadków testowych może spowodować, że informacja o jakości testowanego oprogramowania będzie kompletna i wiarygodna.