Typy testów oprogramowania. Definicja i metody badania użyteczności.

# Typy testów oprogramowania

## Testy modułowe (jednostkowe)

Są przeprowadzane na poziomie aplikacji zbliżonym do kodu źródłowego. Polegają na testowaniu pojedynczych metod, obiektów, modułów lub komponentów. Ich ogromną zaletą jest możliwość pełnej automatyzacji procesu testowania oraz znajdowanie błędów natychmiast po ich pojawieniu się.

## Testy integracyjne

Testy integracyjne to krótko mówiąc, weryfikacja integracji, współdziałania gotowych jednostek kodu, modułów, czy nawet całych systemów i usług. Zakończenie fazy testów integracyjnych powinno pozwolić przejść do fazy testów systemowych, czyli powinno pozwolić traktować wszystkie moduły jako cały, funkcjonalny system, który jest zdolny do poprawnego realizowania konkretnych procesów biznesowych. Tylko przekonanie o spójności systemu daje nam „zielone światło” do testów End-To-End, czyli do testów konkretnych biznesowych procesów w oparciu o cały system. Aby efektywnie wykonywać testy integracyjne, musimy mieć pewność, że dane moduły ( które ze sobą powinny korelować, współpracować) – działają prawidłowo oddzielne – czyli przeszły pozytywnie testy jednostkowe. Dopiero wtedy możliwe jest poprawne testowanie integracji, czyli współpracy dwóch czy większej liczby modułów. Zadaniem testera oprogramowania na poziomie testów integracyjnych, jest również budowanie w zespole świadomości, że każdy nawet niewielki błąd, czy niefortunna zmiana w kodzie w jednym z modułów, może mieć bardzo duże negatywne skutki w innym miejscu aplikacji, gdyż system softwarowy to „system naczyń połączonych” (zestaw modułów, które wzajemnie ze sobą korelują).

* **Wewnętrzne** — dotyczą tylko systemu, który rozwijamy i tutaj testujemy wszelkie integracje między tworzonymi modułami.
* **Zewnętrzne** — to testy, które dotyczą integracji z innymi systemami (niekoniecznie naszymi) oraz z zewnętrznymi usługami, które istnieją na rynku. Oba typy integracji są bardzo ważne, dlatego proces testów integracyjnych powinien być przeprowadzony kompleksowo i z dbałością o szczegóły.

## Testy systemowe

Testy systemowe to testy, których zadaniem jest weryfikacja aplikacji jako całości (produktu) lub jej samodzielnego fragmentu, realizującego jakiś większy zakres projektu. Powyższe zdanie jest bardzo ważne, gdyż przekonałem się, że panuje mylne spojrzenie, że testy systemowe, to tylko testy systemu jako gotowego produktu końcowego. Bardzo ważną rzeczą na etapie testów systemowych, jest to, aby testy przeprowadzać na środowisku testowym, które jest w jak największym stopniu zbliżone do środowiska produkcyjnego, na którym docelowo aplikacja będzie funkcjonować. Dlaczego to jest takie ważne?? Otóż, tylko na takim środowisku możemy zmniejszyć ryzyko niewykrycia błędów, które mogą zostać przeoczone z powodu różnic w specyfice i konfiguracji obu środowisk. Testy systemowe powinny skupiać w sobie zarówno testy funkcjonalne, jak i niefunkcjonalne, gdyż jednym z głównych zadań tych testów jest ocena całościowa, generalne sytemu, bądź jego fragmentu. Na tym etapie testów, możemy wykryć potrzebę zastosowania różnych zaślepek, mocków, stymulatorów, które pozwolą nam zasymulować integrację z innymi systemami czy usługami zewnętrznymi.

Aby solidnie wykonać test systemowe powinniśmy przeprowadzić następujące typy testów:

* testy funkcjonalne
* testy niefunkcjonalne (wydajnościowe)
* testy regresji
* testy konfiguracji
* testy bezpieczeństwa

## Testy akceptacyjne

Testy akceptacyjne to testy wykonywane przez klienta, które mają dać mu informację potrzebną do podjęcia świadomej decyzji o odbiorze bądź odrzuceniu produktu. Zwykle tego typu testy klient realizuje u siebie, przy wykorzystaniu swoich zasobów ludzkich w postaci testerów, potencjalnych użytkowników systemu, lub z przy współpracy z niezależnym zespołem testerskim. Testy akceptacyjne weryfikują system pod kontem zgodności z założeniami i wymaganiami, które zostały postawione przed fazą developmentu. Na tym etapie sprawdza się również wszelkie zgodności z zapisami w umowach oraz z obowiązującymi przepisami prawa. Testy akceptacyjne powinny przejść i zweryfikować wszystkie kryteria zakończenia testów, które powinny zostać spisane w planie testów. Faza testów akceptacyjnych zakończona sukcesem jest podstawą do formalnego rozliczenia kontraktu.

# Definicja i metody badania użyteczności

Obraz zawierający stół

Opis wygenerowany automatycznie

**technika wykorzystywana w projektowaniu zorientowanym na użytkownika**

**„Badania użyteczności to systematyczne śledzenie użytkowników oraz ich wymagań w celu dodania kontekstu i wglądu w proces projektowania UX. Badacze wykorzystują różnorodne techniki, narzędzia oraz metodologię, aby wyciągać wnioski, określać fakty i odkrywać problemy. Ujawniają w ten sposób cenne informacje, które można wykorzystać w procesie projektowania”**

Polegają one na obserwacji tego, jak konkretna osoba korzysta z danego systemu.

## Popularne metody

* Dwie spośród tych metod są ze względu na ich dużą przydatność najczęściej stosowanymi metodami badania użyteczności gotowych produktów lub ich prototypów:
  + ocena heurystyczna
  + testy z udziałem użytkowników, których jednym z rodzajów jest badanie z wykorzystaniem urządzeń do śledzenia wzroku

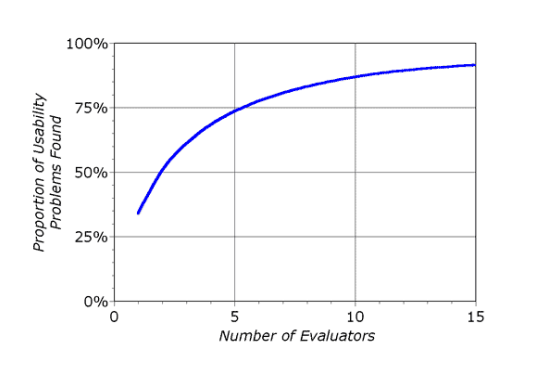
## Testy z udziałem użytkowników

* ocena systemu podczas realizacji przykładowych zadań
* mierzymy
  + skuteczność
  + efektywność
  + satysfakcja użytkowników
* pozwalają zidentyfikować większość problemów użyteczności, także tych który dotyczą specyficznych wymagań bądź umiejętności użytkowników

## Ocena heurystyczna

* Opracowane przez Jacoba Nielsena
* Wymaga zespołu najlepiej 5 ewaluatorów (mniej stronniczy niż członkowie zespołu)
* Wymaga zapoznania z założeniami projektowymi
* Wymaga określenia zbioru heurystyk projektowych

**Liczba ewaluatorów/liczba znalezionych błędów**

****

## 10 heurystyk Nielsena

Obraz zawierający stół

Opis wygenerowany automatycznie

## Inne metody badania użyteczności

1. Badania partyzanckie
2. Testy użyteczności przeprowadzone w laboratorium
3. Niemoderowane zdalne testy użyteczności
4. Wywiad kontekstowy
5. Wywiad telefoniczny
6. Sortowanie kart
7. Nagrywanie sesji
8. Testy 5-sekundowe
9. Testy A/B

## Badania partyzanckie

* Pytanie o zdanie randomowych ludzi na ulicy
* Losowy wybór respondentów
* Szybki test użyteczności wynagrodzony upominkiem
* Niski koszt i prostota
* Najlepiej stosować na wczesnych etapach procesu
* Czas testu jest ograniczony (5-10 min)

## Testy użyteczności przeprowadzone w laboratorium

* Nadzorowane przez moderatora
* Moderatorzy ułatwiają użytkownikom wykonywanie zadań
* Stosowane, gdy potrzeba dogłębnych informacji o napotykanych problemach
* Mała baza uczestników, dlatego ważne jest by byli oni bazą docelową
* Moderator powinien odczytywać mowę ciała
* Debriefing (odprawa, podsumowanie) jest bardzo ważny

## Niemoderowane zdalne testy użyteczności

* Szybkie, solidne i niedrogie
* Wykonywane przy użyciu własnego środowiska użytkownika
* Mniej szczegółowe wyniki niż testy moderowane
* Stosowane gdy potrzeba dużej próbki
* Pozwala obserwować wzorce zachowań użytkowników

## Wywiad kontekstowy

* Pozwala uzyskać informacje o doświadczeniach użytkowników
* Polega na zadawaniu pytań o doświadczenia z produktem
* Następnie obserwuje się pracę użytkownika we własnym środowisku
* Przydatny do uzyskania informacji o użytkownikach (preferencje, nawyki)
* Pozwala na nadanie priorytetu problemom

## Wywiad telefoniczny

* Moderator ustnie instruuje użytkowników
* Informacje zwrotne są zbierane automatycznie
* Interakcja użytkownika jest rejestrowana zdalnie
* Sposób na zebranie opinii od uczestników z całego świata
* Wymaga przeszkolonego moderatora

## Nagrywanie sesji

* Metoda rejestrowania działań anonimowych użytkowników
* Pomaga zrozumieć jakie treści/funkcje są najbardziej interesujące (analiza mapy cieplnej)
* Pomaga zrozumieć główne problemy użytkowników
* Działa najlepiej z innym rodzajem testów użyteczności

## Testy 5-sekundowe

* Zebranie pierwszych wrażeń i reakcji na projekt
* 5 sekund na odpowiedź, przykład:
  + Jakie główne elementy projektu pamiętasz?
* 20-50 uczestników

## Testy A/B

* Porównanie dwóch wersji aplikacji
* Analiza statystyczna pozwala określić, która wersja działa lepiej
* Wykorzystywane w celu optymalizacji witryny
* Im więcej użytkowników, tym lepiej