

Testowanie oprogramowania

Informatyka, sem.5

Rok akademicki 2018/2019

Dr inż. Dominik Olszewski

Literatura:

- R. V. Binder: *Testowanie systemów obiektowych. Modele wzorce i narzędzia*. WNT, Warszawa, 2003. (wyd. ang. 2000).
- G. J. Myers: *Sztuka testowania oprogramowania*. 2005. (wyd. ang. 1979!).
- Różne strony WWW.
- Dokumentacja JUnit, EasyMock.

Jakość oprogramowania

- ***Wszyscy chcemy otrzymywać produkty najwyższej jakości (w tym oprogramowanie).***
- Standard Glossary of Software Engineering Terminology [IEEE610.12] – definicja jakości oprogramowania.
 - Stopień, do którego system, komponent lub proces spełnia wyspecyfikowane wymagania.
 - Stopień do którego system, komponent lub proces spełnia potrzeby lub oczekiwania użytkownika/konsumenta.

Jakość oprogramowania

- *R.S. Pressman: Software Engineering – A Practitioner's Approach (5th ed). 2000.*
- Jakość oprogramowania:
 - Zgodność z jawnie przedstawionymi wymaganiami funkcjonalnymi i wydajnościowymi, jawnie udokumentowanymi standardami wytwarzania, oraz domyślnymi charakterystykami, które są oczekiwane od profesjonalnie utworzonego oprogramowania.

Jakość oprogramowania (3 wymiary)

- Zdolność do działania w innym środowisku:
 - przenośność (ang. portability), możliwość ponownego użycia (ang. reusability), inter-operacyjność (ang. interoperability).
- Podatność na zmiany:
 - Łatwość konserwacji (ang. maintainability), elastyczność (ang. flexibility), łatwość testowania (ang. testability).
- Charakterystyki operacyjne:
 - Poprawność (ang. correctness), niezawodność (ang. reliability), wydajność (ang. efficiency), bezpieczeństwo (ang. integrity), użyteczność (ang. usability).

Jakość oprogramowania – cechy (1)

- Przenośność (ang. portability): Cecha pozwalająca na łatwe przeniesienie wytworzonego systemu z jednego środowiska (sprzęt + oprogramowanie) do innego.
- Możliwość ponownego użycia (ang. reusability): Cecha pozwalająca na ponowne użycie systemu (jego części) w innej aplikacji.
- Inter-operacyjność (ang. interoperability): Cecha pozwalająca na połączenie wytworzonego systemu z innymi systemami.
- Łatwość konserwacji (ang. maintainability): Cecha umożliwiająca wprowadzanie modyfikacji (zwykle poprawek) do systemu.
- Elastyczność (ang. flexibility): Cecha umożliwiająca dokonanie modyfikacji lub dopasowania systemu podczas działania.
- Łatwość testowania (ang. testability): Cecha umożliwiająca przeprowadzenie testów wykazujących poprawność funkcjonowania (działanie funkcji).

Jakość oprogramowania – cechy

(2)

- Poprawność (ang. correctness): Cecha zapewniająca, że system spełnia postawione wymagania i spełnia potrzeby użytkownika.
- Niezawodność (ang. reliability): Cecha powodująca, że system wypełnia swoje zadane funkcjonalności z wymaganą precyzją i bez awarii.
- Wydajność (ang. efficiency): Zasoby systemu są wystarczające do realizacji funkcji (pamięć, czas) .
- Bezpieczeństwo (ang. integrity): Dostęp do systemu i jego danych jest kontrolowany (różne poziomy wymagań).
- Użyteczność (ang. usability): System jest prosty w obsłudze, łatwy do nauczenia się, prosty we wprowadzaniu danych i interpretacji otrzymanych wyników.
- **Jak to zmierzyć – ocenić?**

Zapewnienie jakości (Quality assurance)

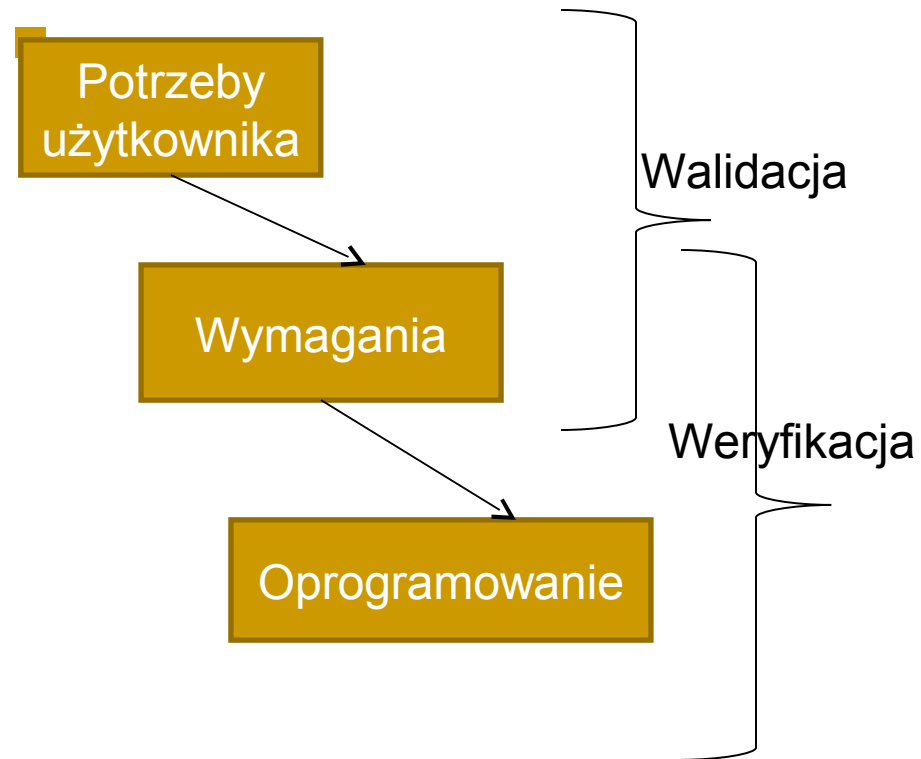
- Standard Glossary of Software Engineering Terminology [IEEE610.12] – definicja zapewnienia jakości (ang. quality assurance).
 - Planowe i systematyczne stosowanie procedur (akcji) zapewniające, że powstały produkt spełnia ustalone wymagania techniczne.
 - Zbiór procedur (akcji) zaprojektowanych dla badania procesu wytwarzania produktu.

Typowe akcje zapewniania jakości

- Stosowanie właściwych metod i narzędzi,
- Przeglądy techniczne i inspekcje,
- Testowanie oprogramowania,
- Stosowanie standardów,
- Tworzenie dokumentacji,
- Kontrola zmian,
- Stałe pomiary,
- Aktualny stan procesu i raportowanie.

Weryfikacja i walidacja

- Weryfikacja: czy produkt jest tworzony zgodnie z wymaganiami (procedurami)?
- Walidacja: czy tworzony produkt jest odpowiedni dla użytkownika?



Testowanie – źródła błędów

- Dlaczego powstaje oprogramowanie z błędami?
 - Łatwość dodawania nowej funkcjonalności,
 - Ambitne cele,
 - Zasada „zawsze zdążę usunąć błędy” – zamiast starannego planowania.
 - Brak „**widocznych**” ograniczeń,
 - Czynniki psychologiczne,
 - Złożoność oprogramowania,
 - Ograniczenia poznawcze,
 - Użytkownicy – możliwe są nieporozumienia i nieprawidłowe definicje (słowniki !),
 - Ograniczenia projektowe: terminy, zasoby, niedostateczna znajomość używanych narzędzi.

Testowanie – koszt błędów

- Koszty błędów w oprogramowaniu:
 - ❑ Gry – koszt zerowy (+ nerwy użytkownika),
 - ❑ Eksplozja rakiety Ariane 5 (1996) – 8 mld USD, zła konwersja dużej liczby.
 - ❑ Satelita marsjański Mars Climate Observer (1999) – 2 mld USD, pomieszanie jednostek wagi (funty i kilogramy),
 - ❑ Strona o skutkach błędów w oprogramowaniu www5.in.tum.de/~huckle/bugse.html
- Utrata zaufania klientów, kontrahentów.

Testowanie – terminologia, słownictwo

- **Słownictwo:** Norma brytyjska BS 7925-1, Software Testing Vocabulary, została przetłumaczona na język polski jako:
- **Słownik wyrażień związanych z testowaniem**
 - http://www.sjsi.org/bin/doc/slownik_v10.html

Testowanie – pojęcia podstawowe

(1)

- **Pomyłka (error):** Działanie człowieka powodujące powstanie nieprawidłowego wyniku.
- **Usterka (fault):** Skutek pomyłki (błędu) twórcy oprogramowania. Usterka może, ale nie musi spowodować awarię.
- **Awaria (failure):** Odchylenie od spodziewanego zachowania albo wyniku działania oprogramowania. Skutki awarii mogą być różnej wagi.

Testowanie – pojęcia podstawowe

(2)

■ Weryfikacja

- “**Proces kontroli systemu lub modułu polegający na sprawdzeniu, czy produkty danego etapu produkcji spełniają warunki zadane na początku tego etapu**”.
 - czyli
- Sprawdzenie czy poprawnie **zbudowano aplikację**
 - czyli
- Aplikacja jest **zgodna z jej specyfikacją**.

Testowanie – pojęcia podstawowe

(3)

■ Walidacja

- ❑ Określenie poprawności produktów procesu tworzenia oprogramowania pod względem potrzeb i wymagań użytkownika
 - ❑ czyli
- ❑ Sprawdzenie czy aplikacja jest **poprawna**
 - ❑ czyli
- ❑ **Spełnione są życzenia klienta.**

Testowanie – cel działania

- Określenie poziomu jakości produktu i jego rzetelności (reliability).
- **Rzetelność (Reliability):**
“prawdopodobieństwo, że aplikacja nie spowoduje awarii w **określonym czasie przy zadanych warunkach**”
- **Walidacja i weryfikacja to cele** procesu testowania, a nie jego etapy.