

Testowanie i weryfikacja oprogramowania

Wymiar przedmiotu: **5 punkty**

Forma zaliczenia: **Egzamin**

Cel przedmiotu

Testowanie jest jednym z etapów tworzenia oprogramowania. Nie istnieją formalne metody stwierdzenia, że wytworzone oprogramowanie spełnia postawione mu wymagania. Testy mogą być przeprowadzane na niskim poziomie (testowanie poszczególnych metod klas), poziomie modułów, pakietów, aż do poziomu testów akceptacyjnych całego systemu. Prezentowane różne techniki testowania testowanie czarno-skrzynkowe, białe skrzynkowe i integracyjne. W czasie laboratorium studenci będą mogli zapoznać się z oprogramowaniem JUnit.

Treść przedmiotu

1. Wprowadzenie.

- Przyczyny powstawania błędów.
- Koszt błędów oprogramowania.
- Podstawowe definicje pomyłka (ang. error), usterka (ang. fault), awaria (ang.failure), weryfikacja, walidacja. Testowanie kompletne (gruntowne).
- Rodzaje testów – czarno-skrzynkowe, białe-skrzynkowe.

2. Przypadek testowy.

- Wyrocznia (ang. oracle).
- Testowanie dziedziczne.
- Klasy równoważności, analiza wartości brzegowych.
- Wybór przypadków testowych. Przejścia między stanami.

3. Testowanie według scenariusza.

- Zasady budowy scenariusza.
- Model V.

4. Testy modułowe, pokrycie testowe.

- Testy wiodące oprogramowanie (ang. test driven development).
- Test integracyjne w małej skali, sposoby ich prowadzenia.

5. Testy systemowe.

- Niezależny zespół prowadzący testy.
- Testy wymagań funkcjonalnych, testy procesów biznesowych.
- Testy integracyjne dużej skali (środowisko produkcyjne).
- Testy akceptacyjne (alfa, beta).
- Testy regresywne.
- Obiekty pozorne.

6. Planowanie testów.

- Planowanie niezbędnych zasobów.
- Szacowanie pracochłonności.
- Harmonogram testów.
- Dokumentacja testów.
- Krytyczność (powaga) błędu.
- Zarządzanie zgłoszeniami błędów.

7. Kontrola postępu prac testowych.

- Konfiguracja środowiska.
- Testy przeciążeniowe.
- Testy wydajnościowe.
- Testy zabezpieczeń.
- Testy użyteczności.
- Testy zgodności z innymi systemami.