

Szkolenie "Testowanie automatyczne" - oferta

Plan szkolenia

Dzień 1

SOLID

1. Co to jest SOLID?
2. Zastosowanie SOLID w praktyce
3. Dodatkowe dobre praktyki
4. Dependency Injection

Testy jednostkowe

1. Co to są i po co są testy?
2. Jak pisać testy
3. Test Driven Development (TDD)

Dzień 2

1. Mocki
2. Praktyczne scenariusze wykorzystania testów
3. Biblioteki i narzędzia - przegląd i rekomendacje
4. Testy bazy danych
5. Testowanie scenariuszy wielowątkowych
6. Testowanie integracji z usługami zewnętrznymi

Dzień 3

1. Convention tests
2. Behavior Driven Development (BDD)
3. Testowanie mutacyjne
4. Dodatkowe ćwiczenia praktyczne
5. Testowanie ASP.NET MVC i JavaScript

Opis szkolenia

Trzydniowe szkolenie poruszające wszelkie możliwe aspekty tworzenia testów jednostkowych. Oprócz testów jednostkowych omówione zostają również wybrane inne typy testów oraz różne podejścia do ich tworzenia i utrzymywania.

Zaczynamy od omówienia i przećwiczenia w praktyce zasad SOLID oraz innych strategii, dzięki którym **kod staje się czytelniejszy i łatwiejszy w utrzymaniu**. Podczas zadań praktycznych zobaczymy jakie kroki pozwalają na refaktoryzację istniejącego kodu do "lepszego" postaci... bo przecież większość pracy programisty polega na utrzymywaniu i modyfikacji już zaimplementowanych rozwiązań. Kod "lepszy" prawie zawsze implikuje również charakterystykę: kod "łatwo testowalny".

Po zapoznaniu się/utrwaleniu najistotniejszych reguł programowania obiektowego intensywnie zagłębiamy się w temat testów jednostkowych. Uczestnicy poznają zarówno najlepsze praktyki dotyczące tego zagadnienia, jak również różne **korzyści płynące z pokrycia kodu testami automatycznymi**. Wbrew pozorom zredukowana liczba błędów w finalnym oprogramowaniu jest tylko jednym z wielu celów. Świadome - pragmatyczne, a nie dogmatyczne - zastosowanie testów

pomaga pisać przyjaźniejszy kod oraz, co istotne - zwiększyć satysfakcję programisty z wykonywanej pracy. Podczas dyskusji oraz wykonywanych zadań trener dzieli się swoim wieloletnim doświadczeniem, spostrzeżeniami oraz wypracowanymi sposobami na możliwe bezbolesne rozpoczęcie bądź rozwinięcie przygody uczestników z testami. Pierwszego dnia poruszamy również temat **Test Driven Development (TDD)** - jednej z popularnych metod pisania testów jednostkowych.

Drugiego dnia skupiamy się głównie na praktycznych aspektach testowania. Poznajemy pojęcie "**mocków**" i podczas wielu zadań badamy jak testy sprawdzają się w sytuacjach, które każdy programista spotyka na co dzień. Omawiamy również bardzo szeroki **wachlarz dostępnych na rynku bibliotek i narzędzi**. Druga część dnia to dogłębne i szczegółowe przećwiczenie przysparzających trudności scenariuszy: testowanie **baz danych**, komponentów działających w **środowisku wielowątkowym** oraz **integracji tworzonych systemów z usługami zewnętrznymi**.

Dzień trzeci to głównie poszerzanie horyzontów, mające na celu wskazanie dodatkowych zastosowań testów ("**convention tests**") oraz sposobów ich tworzenia ("**Behavior Driven Development - BDD**"). Wisienką na testowym torcie jest przybliżenie uczestnikom szkolenia koncepcji "**testowania mutacyjnego**", o której niewiele osób słyszało. Końcówka materiału szkoleniowego to kilka dodatkowych ćwiczeń utrwalających zdobytą podczas trzech dni wiedzę. Następnie poświęcimy kilkadziesiąt minut na testowanie frameworka **ASP.NET MVC** oraz kodu **JavaScript**, co często jest najbardziej zaniedbanym obszarem podczas pokrywania oprogramowania testami.

Niniejsze szkolenie to w ok **80% ćwiczenia praktyczne**, dzięki którym już następnego dnia uczestnicy są w stanie wykorzystać zdobytą wiedzę w codziennej pracy. Części teoretyczne są uzupełnieniem ćwiczeń oraz punktem wyjściowym do aktywnej dyskusji z uczestnikami.

Każdy z trzech dni podzielono na pięć lub sześć sekcji trwających około 60-90 minut. Pozostały czas przeznaczony jest na **dyskusje trenera z zespołem**, wymianę doświadczeń oraz wspólnie poszukiwanie najlepszych rozwiązań.

Szkolenie kierowane jest do **programistów .NET**, gdyż ćwiczenia oraz prezentowane narzędzia powiązane są z tym właśnie środowiskiem.

W trakcie szkolenia uczestnicy będą wykorzystywać **system kontroli wersji Git**, co w przypadku nieznamości tego narzędzia może być dodatkowym walorem edukacyjnym.

Maksymalna liczba uczestników: **10**.

Trener: Maciej Aniserowicz
maciej.aniserowicz@gmail.com
<http://devstyle.pl>
<http://devtalk.pl>

Wybrane opinie dotychczasowych uczestników szkolenia

- "Nie było rozgraniczenia na zasadzie 'guru vs uczniowie'. Widać było, że trener jest praktykiem, więc łatwo był w stanie przytaczać realne sensowne przykłady z życia zamiast bazować tylko na teorii."
- "Bardzo szeroki przekrój rodzajów testów, o kilku rodzajach testów nie słyszałem."
- "Bardzo podobało mi się zestawienie teorii z zastosowaniem w praktyce, łącznie z przeglądem bibliotek."
- "Bardzo pozytywnie odbieram to szkolenie. Nabyłem dużą ilość wiedzy, którą będę umiał zastosować praktycznie."
- "Bardzo dobre szkolenie. Wszystko przedstawione w sposób przystępny."
- "Dobra proporcja teorii, do ćwiczeń i do dyskusji."
- "Bardzo ważne jest to, że trener miał dystans do siebie i akceptował to, że możemy mieć inne

doświadczenia. Był otwarty na inne podejścia i dyskusje."

- "Konkretne, rzetelnie poprowadzony warsztat. Bardzo dobrze dobrana ilość materiału, dobra organizacja czasu."
- "Szkolenie pozwoliło na lepsze uporządkowanie już posiadanej wiedzy i pokazało pewne tematy z innej perspektywy."
- "Szkolenie użyteczne, porządkujące wiedzę."
- "Spełniło moje oczekiwania, otworzyło oczy na kilka tematów, które teraz wydają się oczywiste."
- "Bardzo owocne"
- "Pozwoliło na usystematyzowanie wiedzy oraz zainspirowało do zastanowienia się na jakością pisanych testów."