

Szkolenie "Warsztat programisty" - oferta

Plan szkolenia

Dzień 1 - SOLID

1. Co to jest SOLID?
2. Zastosowanie SOLID w praktyce
3. Dodatkowe dobre praktyki
4. Dependency Injection
5. Autofac - przykładowy kontener Dependency Injection
6. Dependency Injection w ASP.NET MVC (na przykładzie Autofac)

Dzień 2 - testy jednostkowe

1. Co to są i po co są testy?
2. Jak pisać testy
3. Test Driven Development (TDD)
4. Mocki
5. Praktyczne scenariusze wykorzystania testów
6. Biblioteki i narzędzia - przegląd i rekomendacje

Dzień 3 - CQRS (Command Query Responsibility Segregation) + microservices

1. CQRS - wstęp (prezentacja konferencyjna)
2. CQRS - praktyka - READ
3. CQRS - praktyka - WRITE & testing
4. Events
5. Microservices
6. Deployment

Opis szkolenia

Trzydniowe szkolenie poruszające bardzo ważne kwestie z punktu widzenia programisty.

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 "lepszej" postaci... bo przecież większość pracy programisty polega na utrzymywaniu i modyfikacji już zaimplementowanych rozwiązań.

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.

Na koniec przechodzimy do podejścia CQRS (Command/Query Responsibility Segregation) oraz ewolucji do architektury mikroserwisów. Te pojęcia, zaaplikowane w odpowiednich scenariuszach, pozwalają na **zmaksymalizowanie wydajności zespołów programistycznych, jednocześnie minimalizując narzut związany z utrzymywaniem wiekowego oprogramowania**. Niezmiernie istotne jest jednak uświadomienie sobie, że mimo bieżącej "mody" stosowanie CQRS czy mikroserwisów nie zawsze ma sens. Należy również pochylić się nad nowymi problemami wynikającymi z ich wykorzystania oraz możliwymi sposobami ich rozwiązywania.

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 podzielony jest na sześć sekcji trwających około godziny. Pozostały czas przeznaczony jest na **dyskusje trenera z zespołem**, wymianę doświadczeń oraz wspólne 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://www.maciejaniserowicz.com>
<http://devtalk.pl>

Wybrane opinie dotychczasowych uczestników szkolenia

- "Super szkolenie"
- "Dla mnie było to najlepsze szkolenie od dłuższego czasu. Bardzo dobrze przygotowane."
- "Większość czasu była poświęcona zadaniom praktycznym, czyli programowaniu - moim zdaniem tak właśnie powinno być"
- "Bardzo fajne i przydatne"
- "Dobrze wyważone szkolenie"