

# Inversion of Control

---

## CZAS TRWANIA

2-3 dni

## ABSTRAKT

Szkolenie ma na celu pokazanie możliwości technik odwracania kontroli.

## ZAGADNIENIA

### Statyczność czy instancyjność

Dlaczego statyczność jest problematyczna?  
Czy instancyjność jest wolna?  
Testowanie

### Dobre praktyki

Dependency Inversion Principle  
Low coupling i high cohesion

### Singleton

Jak zaimplementować poprawnie wzorzec?  
Statyczny konstruktor w .NETcie  
Dlaczego używanie singletonu w większości wypadkach jest antywzorcem?  
Kiedy używać singleton?

### ServiceLocator

Implementacja wzorca Mediatora, który separuje abstrakcje od implementacji  
Problemy podejścia

### Inversion of Control - IoC

Dependency Injection

Jeden ze sposobów odwracania kontroli. Wysokopoziomowe komponenty zależą od abstrakcji.

Ręczne podejście do DI

Kiedy kontener DI staje się ServiceLocatorem?

Jak napisać kod, który nie ma zależności do żadnego kontenera?

Użycie DI w następujących typach aplikacji

WinForms

Wzorzec Navigatora

WPF

Asp.Net WebForms

Asp.Net MVC

WCF Service

Przydatne wzorce projektowe

Fabryka jako obiekt enkapsulujący tworzenie konkretnych implementacji komponentów

Castle Windsor jako kontener DI

Automatyczne składanie wzorca dekoratora i chain of responsibility przez kontener

Kiedy używać rejestracji w kodzie, a kiedy w konfiguracji xml.

Jak dynamicznie zmieniać zależności?

Jak wielokrotnie tworzyć subkomponent podczas życia długotrwałego obiektu bez zależności do komponentu?

TypedFactory

Castle Facilities

Architektura SaaS, a kontener DI

Wielopoziomowe kontenery DI

Testowanie kodu wspierającego kontener DI

## Zdarzenia

Wzorzec Observer

Implementacja .NETowa i jej problemy

EventBroker/EventAggregator

Luźny sposób łączenia komponentów

RPC vs Messaging

Implementacje

Działanie w procesie w obrębie transakcji

Wstęp do szyny danych

Zastosowania

Integracja modułów aplikacji - autonomicznych komponentów

Audyt, logowanie

Tworzenie projekcji modelu biznesowego na model tylko do odczytu

Jak stworzyć model w 1NF z modelu w 3NF

Projektory i Data Transfer Object

Separacja infrastruktury od Core Biznesowego

Wysyłanie e-maili, sms, poczty

Chain of responsibility

EventSourcing

Przykład modelu persistencji opartego o strumień zdarzeń

Interfejs użytkownika

Komunikacja partów interfejsowych

Integracja z logiką biznesową

Długotrwałe procesy biznesowe

Saga, Workflow

Procesy składające się z sekwencji wielu zdarzeń

Testowanie

Jak wystestować reakcję na zdarzenia?

## Programowanie aspektowe

Możliwości funkcjonalne i нефункционалне aplikacji

Cross-cutting concerns

Programowanie funkcyjne jako rozwiązanie połowiczne

Delegaty i Lambdy w .NETcie

Postsharp jako biblioteka wspierająca AOP w .NETcie

Tryb działania i wstrzykiwania aspektów

Rodzaje aspektów

Przykładowe aspekty

CacheAspect

OnThreadPoolAspect

ImplementNotifyPropertyChangedAspect

TraceAspect i DebugAspect

ProfileAspect

HandleExceptionsAspect

Aspekty w Castle

Interceptory  
Wybieranie metod do przechwycenia

**KONTAKT**

W celu omówienia szczegółów i rezerwacji terminu, skontaktuj się z nami:

**[kontakt@macmichal.pl](mailto:kontakt@macmichal.pl)**

**tel. 513 959 379**

