

# Tytuł szkolenia: Tworzenie usług sieciowych REST API w technologii ASP.NET WebAPI

Kod szkolenia: MS-REST

## Wprowadzenie

Wzrost popularności urządzeń mobilnych z dostępem do sieci spowodował zapotrzebowanie na tworzenie rozwiązań z zastosowaniem usług sieciowych.

**WebAPI** to środowisko, które umożliwia szybkie i łatwe budowanie usług sieciowych, które później mogą być dostępne z aplikacji webowych lub urządzeń mobilnych.

Szkolenie dotyczy **.NET Framework 4.7**

Jeśli chcesz poznać wersję opartą **.NET Core** - zapraszamy na szkolenie [.NET 5](#).

## Adresaci szkolenia

Grupa docelowa szkolenia to głównie **osoby pełniące kluczowe aktywności przy realizacji projektu** już w samym zespole projektowym. Adresatami szkolenia są **techniczni liderzy zespołów**, jak również **programiści**. Znajomość podstawowych zwrotów z zakresu rozwoju oprogramowania. Bardzo zalecane uczestnictwo w szkoleniu [MS-C#-5-VS] [Programowanie w C# 7.0 z Visual Studio 2017](#) (ang. Programming C# 7.0 with Visual Studio 2017).

Na szkoleniu wymagana jest znajomość programowania w **języku C#** z użyciem Visual Studio 2010 (lub wyżej), znajomość SOAP, WCF będzie dodatkowym atutem.

## Cel szkolenia

Celem szkolenia jest **nabywanie umiejętności tworzenia usług sieciowych** z zastosowaniem **WebAPI** i korzystania z nich w aplikacjach desktopowych oraz mobilnych w Windows 10.

W trakcie kursu uczestnicy zapoznają się również z **zasobami MSDN**, a także narzędziami wspierającymi rozwój oprogramowania, takimi jak **Microsoft Visual Studio**.

## Czas i forma szkolenia

- 21 godzin (3 dni x 7 godzin), w tym wykłady i warsztaty praktyczne.

## Plan szkolenia

### Wprowadzenie do WebAPI

- a. Porównanie XML Web Services, WCF i WebAPI
- b. Porównanie WebAPI i ASP.NET MVC

### Utworzenie usługi WebAPI

- a. Utworzenie kontenera
- b. Obsługa HTTP
- c. Konfiguracja routingu
- d. Routing za pomocą atrybutów
- e. Definiowanie reguł parametrów

### RESTfull

- a. Projektowanie REST API
- b. Obsługa błędów

### Przetwarzanie komunikatów

- a. Zasada działania
- b. Architektura cebuli
- c. Tworzenie własnego handlera
- d. Logowanie komunikatów

### Negocjowanie formatu

- a. Zasada działania
- b. Tworzenie obsługi własnego formatu
- c. Przekazywanie formatu w adresie URL

### Walidacja

- a. Walidacja za pomocą atrybutów
- b. Walidacja za pomocą biblioteki FluentValidation

### Bezpieczeństwo

- a. Podstawowa autentykacja
- b. Autentykacja oparta o tokeny

### Hostowanie

- a. IIS
- b. Self-hosting
- c. OWIN

### Tworzenie klienta REST API

- a. Tworzenie klienta z użyciem HttpClient

### Dokumentowanie API

- a. Utworzenie dokumentacji w formacie Swagger/OpenAPI
- b. Generowanie klienta na podstawie formatu Swagger

### Narzędzia

- a. Postman
- b. ngrok