Kod szkolenia: **NODE/TS**

Tytuł szkolenia: **Nowoczesny Backend z użyciem Node.js, TypeScript i frameworka NestJS**

Modern Backend with Node.js, TypeScript and NestJS Framework

Dni: 4

Opis:

Adresaci szkolenia

Szkolenie jest przeznaczone dla programistów ze znajomością JavaScript w zakresie podstawowym. Którzy chcą poznać nowoczesne metody budowy aplikacji serwerowych w technologi Node.js i TypeScript.

Szkolenie jest specjalnie polecane dla programistów Angular, którzy by chcieli zacząć pisać backend, gdyż architektura NestJS jest mocno nim inspirowana.

Cel szkolenia

Głównym celem szkolenia jest przekazanie praktycznej wiedzy pozwalającej tworzyć aplikacje serwerowe w oparciu o Node.js i TypeScript, a w tym:

- Poznasz dobre praktyki, wzorce architektoniczne i narzędzia pozwalające na budowanie aplikacji, które będą skalowalne oraz łatwe w utrzymaniu i rozwoju.
- Poznasz architekturę Node.js oraz frameworka Express.js.
- Porównasz aplikację napisaną w czystym JavaScript oraz napisaną w TypeScript.
- Poznasz nowoczesny framework NestJS i porównanie jego zalet na tle aplikacji opartej o Express.
- Poznasz podstawy TypeScript na praktycznych przykładach.
- Zbudujesz REST API za pomocą NestJS wraz z automatycznie generowaną dokumentacją w formacie Swagger.
- Połączysz się z baza danych SQL przy użyciu TypeORM.
- Wykorzystasz i wzbogacisz swoją wiedzę tak, by stworzyć skalowalną aplikację Node.js w NestJS
- Nauczysz się jak pisać i uruchamiać testy jednostkowe oraz e2e.
- Uruchomimy logikę biznesową naszej aplikacji w różnych kontekstach, takich jak REST API, WebSocket, RPC (mikroserwisy), wiersz poleceń (CLI).

Na koniec poznamy proces budowania, wgrywania, uruchamiania i monitorowania naszej

aplikacji na produkcji.

Mocne strony szkolenia

Szkolenie skupia się na tworzeniu aplikacji, które będą uruchamiane jako usługa lub w kontenerze Dockera.

Przekazana wiedza będzie pozwalała tworzyć oprogramowanie zgodne z ideą The Twelve-Factor App.

Wymagania

Od uczestników wymagana jest podstawowa znajomość JavaScript (ES2017). Zagadnienia, które należy znać: var, const, let, pętle, operacje warunkowe, funkcja, arrow function, klasa, ES Modules, Promise, async/await

Specjalne wymagania techniczne

Wymagany jest komputer z zainstalowanym Node.js w minimalnej wersji 6.11.0 (zalecana wersja >= 10) i edytorem wspierającym TypeScript (zalecany to darmowy Visual Studio Code)

Parametry szkolenia

4*8 godzin (4*7 godzin netto) wykładów i warsztatów (z wyraźną przewagą warsztatów).

Program szkolenia:

1. Wprowadzenie
    - Node.js
        - Jednowątkowa czy wielowątkowa architektura
        - Nieblokujące operacje wejścia/wyjścia
        - Moduły CommonJS
    - Node Package Manager (npm)
    - Asynchroniczność
        - Callback
        - Promise
        - async/await
        - Observable
    - TypeScript
        - ES Modules
        - Proces kompilacji TS do JS
        - Statyczne typowanie
        - Interfejsy
        - Dekoratory

        - Typy generyczne
2. Express
    - Budowa prostego serwera API
    - Routing
    - Architektura middleware
3. Architektura NestJS
    - Nest CLI
        - Generowanie nowej aplikacji
        - Generowanie komponentów aplikacji
    - Dependency Injection
    - Module
    - Controller
    - Providers
    - Service
    - Guard
    - Middleware
    - Custom Decorators
    - Pipes
    - Interceptors
    - Exception Filter
4. Budowa REST API
    - Routing
    - Autoryzacja
    - Walidacja
    - Upload plików
    - Generowanie dokumentacji Swagger
    - Serwowanie HTML i plików statycznych
    - Konfiguracja
5. Bazy danych
    - TypeORM
    - SQLite/MySQL
    - Migracje struktury bazy danych
6. Testowanie
    - Testy e2e
        - Zapytania: GET, POST, DELETE...
        - Upload plików
    - Testy jednostkowe
        - Testowy moduł
        - Mockowanie serwisów
    - Automatyczne tworzenie danych testowych (fixtures)
        - Pliki i struktura katalogów
        - Uzupełnianie bazy testowymi rekordami
7. Inne konteksty wywołania
    - CLI
        - uruchamianie elementów aplikacji z linii poleceń
        - uruchamianie zadań CRON

- WebSocket - komunikacja real-time
- RPC - Mikroserwisy

8. Build, deploy i monitoring aplikacji
    - Budowa skryptów budujących z shell.js
    - Narzędzia do deploymentu
    - Monitoring i zarządzanie procesami z PM2