

Kod szkolenia: **DED/FSJS**

Tytuł szkolenia: **Full Stack JavaScript z Angular i Nest**

Dni: 5

Opis:

Adresaci szkolenia

Kurs przeznaczony jest dla programistów posiadających podstawową wiedzę w zakresie JavaScript, którzy chcą poznać nowoczesne poszerzyć swoje kompetencje związane z tworzeniem zaawansowanych aplikacji internetowych działających po stronie przeglądarki. Oraz którzy chcą poznać nowoczesne metody budowy aplikacji serwerowych w technologii Node.js i TypeScript.

Cel szkolenia

Celem szkolenia jest zapoznanie uczestników z dobrymi praktykami programowania w języku JavaScript oraz praktycznymi zagadnieniami związanymi z tworzeniem aplikacji internetowych w architekturze "Single-page application" przy użyciu frameworka Angular.

Osoby, które ukończą szkolenie dowiedzą się jak:

- konstruować czytelne i wydajne aplikacje w JavaScript oraz TypeScript z wykorzystaniem aktualnych możliwości tych języków,
- używać elementów programowania funkcyjnego i reaktywnego do rozwiązywania typowych problemów,
- prawidłowo i efektywnie skonfigurować środowisko do pracy z zaawansowanymi projektami front-endowymi,
- stosować aplikacje z oparciem o kontrolę jakości kodu aplikacji JavaScript,

Powyższa wiedza umożliwi uczestnikom samodzielne tworzenie zaawansowanych aplikacji webowych. Głównym celem drugiej części 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 bazą 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.

Program szkolenia:

Program

1. JavaScript i TypeScript

- I. Programowanie reaktywne z RxJS
- II. ECMAScript 2015
 - i. Narzędzia do transpilacji
 - ii. Podstawowe struktury
 - iii. Praktyczne zastosowanie
 - iv. Linting kodu ES2015
- III. TypeScript
 - i. Wprowadzenie do TypeScript
 - ii. Migracja na TypeScript
 - iii. Definicje typów dla zewnętrznych bibliotek
 - iv. Linting TypeScript

1. Angular

- I. Angular a AngularJS - różnice i cechy wspólne
- II. Architektura Angular
 - i. Elementy składowe frameworka

ii. Wstrzykiwanie zależności w Angular

III. Startowanie aplikacji

IV. Komponenty

i. Proste komponenty

ii. Komponenty zagnieżdżone

iii. Przekazywanie parametrów do komponentu

iv. Metody komunikacji między komponentami

v. Cykl życia komponentów

V. Warstwa widoku

i. Elementy składni szablonów

ii. Poszczególne rodzaje wiązania danych

iii. Dyrektywy strukturalne

VI. Formularze

i. Budowa formularzy

ii. Stany pól formularza

iii. Obsługa walidacji

iv. Własne reguły walidacji

VII. Komunikacja z RESTowym API

i. Wprowadzenie do modułu HttpClient

ii. Metody i dodatkowa konfiguracja zapytań

VIII. Usługi

i. Organizacja danych w aplikacji

ii. Wykorzystanie RxJS

IX. Filtry

i. Filtrowanie danych za pomocą Pipe

ii. Przekazywanie parametrów do fitru

iii. Filtry stanowe i asynchroniczne

X. Routing

i. Idea routingu w aplikacjach SPA

ii. Podstawy Component Routera

iii. Routing strategies

iv. Przekazywanie parametrów w URL

v. Routing zagnieżdżony

vi. Cykl życia w routingu i jego praktyczne zastosowanie

XI. Testy jednostkowe elementów składowych aplikacji

i. Testowanie filtrów

ii. Testowanie usług

iii. Mockowanie komunikacji http

iv. Testowanie komponentów

v. Testowanie formularzy

1. Architektura NestJS

I. Nest CLI

i. Generowanie nowej aplikacji

ii. Generowanie komponentów aplikacji

II. Dependency Injection

III. Module

IV. Controller

V. Providers



- VI. Service
- VII. Guard
- VIII. Middleware
- IX. Custom Decorators
- X. Pipes
- XI. Interceptors
- XII. Exception Filter

1. Budowa REST API

- I. Routing
- II. Autoryzacja
- III. Walidacja
- IV. Upload plików
- V. Generowanie dokumentacji Swagger
- VI. Serwowanie HTML i plików statycznych
- VII. Konfiguracja

1. Bazy danych

- I. TypeORM
- II. SQLite/MySQL

1. Testowanie

I. Testy e2e

i. Zapytania: GET, POST, DELETE...

ii. Upload plików

II. Testy jednostkowe

i. Testowy moduł

ii. Mockowanie serwisów

