



tbprojects Tomasz
Borowski

★★★★★ 4,6 / 5

71 ocen

Angular Master - zaawansowane szkolenie dla programistów aplikacji internetowych w technologii Angular

Numer usługi 2025/12/18/17242/3221827

📍 zdalna w czasie rzeczywistym

🏠 Usługa szkoleniowa

🕒 24 h

📅 25.03.2026 do 27.03.2026

3 196,77 PLN brutto

2 599,00 PLN netto

133,20 PLN brutto/h

108,29 PLN netto/h

Informacje podstawowe

Kategoria	Informatyka i telekomunikacja / Programowanie
Identyfikatory projektów	Małopolski Pociąg do kariery, Nowy start w Małopolsce z EURESEM
Grupa docelowa usługi	Szkolenie w wariantcie Angular Master przeznaczone jest dla programistów stron lub aplikacji internetowych, którzy znają już podstawy frameworka Angular i towarzyszących mu technologii oraz chcą poznać techniki zaawansowane. Każdy uczestnik powinien posiadać podstawową wiedzę z zakresu programowania w Angular oraz potrafić tworzyć proste aplikacje w tym frameworku. Usługa również adresowana dla Uczestników Projektu MP i/lub dla Uczestników Projektu NSE
Minimalna liczba uczestników	4
Maksymalna liczba uczestników	25
Data zakończenia rekrutacji	24-03-2026
Forma prowadzenia usługi	zdalna w czasie rzeczywistym
Liczba godzin usługi	24
Podstawa uzyskania wpisu do BUR	Standard Usługi Szkoleniowo-Rozwojowej PIFS SUS 2.0

Cel

Cel edukacyjny

Szkolenie przygotowuje do samodzielnej implementacji zaawansowanej aplikacji internetowej, przy wykorzystaniu technologii Angular, PWA, NgRx oraz testowania aplikacji przy wykorzystaniu technologii Cypress.

Szkolenie umożliwia uczestnikom nabycie kwalifikacji istotnych dla transformacji cyfrowej, w szczególności w zakresie projektowania i wdrażania aplikacji internetowych wspierających cyfryzację procesów biznesowych oraz unowocześnianie usług i produktów z wykorzystaniem technologii webowych.

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Posługuje się wiedzą dotyczącą frameworka Angular	wskazuje charakterystyczne cechy architektury monorepo oraz bibliotek Angular	Test teoretyczny z wynikiem generowanym automatycznie
	wskazuje przeznaczenie usług typu Component Provider	Test teoretyczny z wynikiem generowanym automatycznie
	wskazuje zasady działania kombinacyjnych i mapujących operatorów RxJS	Test teoretyczny z wynikiem generowanym automatycznie
	wskazuje przeznaczenie poszczególnych technik optymalizacyjnych	Test teoretyczny z wynikiem generowanym automatycznie
	wskazuje rozwiązania odpowiedzialne za dynamiczne renderowanie treści	Test teoretyczny z wynikiem generowanym automatycznie
	wskazuje możliwe do rejestracji wartości podczas definiowania InjectionTokenów	Test teoretyczny z wynikiem generowanym automatycznie
	wskazuje przeznaczenia stosowania Http Interceptorów oraz App Initializerów	Test teoretyczny z wynikiem generowanym automatycznie
	wskazuje elementy związane z dynamiczną modyfikacją formularzy	Test teoretyczny z wynikiem generowanym automatycznie
	wskazuje charakterystyczne cechy aplikacji progresywnych	Test teoretyczny z wynikiem generowanym automatycznie
	wskazuje kluczowe elementy związane z zarządzaniem danymi z NgRx	Test teoretyczny z wynikiem generowanym automatycznie
Tworzy i wykorzystuje biblioteki Angular w aplikacji	implementuje wyświetlanie komponentu pochodzącego z utworzonej biblioteki	Analiza dowodów i deklaracji
Tworzy komponenty prezentacyjne w architekturze smart / dumb components	implementuje komponent komunikujący się za pomocą mechanizmu input / output	Analiza dowodów i deklaracji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Przygotowuje zestawy danych na podstawie Angular Signals	implementuje mechanizm odpowiedzialny za filtrowanie i sortowanie danych	Analiza dowodów i deklaracji
Profiluje i optymalizuje renderowanie komponentów	implementuje wyświetlanie treści dopasowanych ilością i rozmiarem do widoku	Analiza dowodów i deklaracji
Wykorzystuje mechanizm App_INITIALIZER do pobierania danych podczas uruchamiania aplikacji	implementuje funkcję pobierającą i wyświetlającą dane dotyczące konfiguracji aplikacji	Analiza dowodów i deklaracji
Zabezpiecza dostęp do chronionych funkcjonalności z użyciem Guardów	implementuje mechanizm sprawdzający i informujący o braku uprawnień do widoku	Analiza dowodów i deklaracji
Wykorzystuje dyrektywy strukturalne do warunkowego renderowania treści	implementuje wyświetlanie elementów na podstawie konfiguracji aplikacji	Analiza dowodów i deklaracji
Definiuje mechanizmy reaktywnego przetwarzania danych	implementuje mechanizm reagowania na zmiany wartości formularza	Analiza dowodów i deklaracji
Wykorzystuje mechanizm odroczonego ładowania treści	implementuje dynamiczne ładowanie i renderowanie komponentu	Analiza dowodów i deklaracji
Tworzy komponenty dla złożonych kontrolki formularza	implementuje komponent pozwalający na wybór opcji za pomocą zależnych list wyboru	Analiza dowodów i deklaracji
Wprowadza dynamiczne zmiany w strukturze i logice formularza	implementuje dodawanie i usuwanie zagnieżdżonych elementów formularza	Analiza dowodów i deklaracji
Stosuje dynamiczne renderowanie do rozszerzania zawartości widoku	implementuje dynamiczne renderowanie pól formularza wg jego zmiennej konfiguracji	Analiza dowodów i deklaracji
Przygotowuje wyświetlanie widoków z wykorzystaniem routingu drugorzędowego	implementuje wyświetlanie szczegółowych danych w niezależnym widoku	Analiza dowodów i deklaracji
Tworzy mechanizm pozwalający na przechwytywanie zapytań i odpowiedzi API	implementuje funkcję śledzącą aktywność związaną z komunikacją z API	Analiza dowodów i deklaracji
Zarządza globalnie danymi aplikacji z wykorzystaniem biblioteki NgRx	implementuje zestaw funkcji odczytujących i zmieniających stan danych aplikacji	Analiza dowodów i deklaracji

Kwalifikacje

Kompetencje

Usługa prowadzi do nabycia kompetencji.

Warunki uznania kompetencji

Pytanie 1. Czy dokument potwierdzający uzyskanie kompetencji zawiera opis efektów uczenia się?

TAK

Pytanie 2. Czy dokument potwierdza, że walidacja została przeprowadzona w oparciu o zdefiniowane w efektach uczenia się kryteria ich weryfikacji?

TAK

Pytanie 3. Czy dokument potwierdza zastosowanie rozwiązań zapewniających rozdzielenie procesów kształcenia i szkolenia od walidacji?

TAK

Program

Szkolenie przeznaczone jest dla programistów stron lub aplikacji internetowych, którzy znają już podstawy frameworka Angular i towarzyszących mu technologii oraz chcą poznać techniki zaawansowane. Każdy uczestnik powinien posiadać podstawową wiedzę z zakresu programowania w Angular oraz potrafić tworzyć proste aplikacje w tym frameworku.

Szkolenie jest prowadzone w formie zdalnej w czasie rzeczywistym. Uczestnicy sami zapewniają sobie niezbędny sprzęt komputerowy, łącze internetowe oraz oprogramowanie do implementacji aplikacji szkoleniowej. Szkolenie prowadzone jest w trybie godzin zegarowych z przerwami wliczonymi w czas trwania usługi. Szkolenie trwa 24 godziny, na które składa się 10 godzin zajęć teoretycznych i 14 godzin zajęć praktycznych.

W trakcie szkolenia uczestnicy rozwiązują praktyczne zadania, za które otrzymują punkty. Na zakończenie szkolenia uczestnicy rozwiązują teoretyczny test wiedzy. Aby zakończyć szkolenie z wynikiem pozytywnym uczestnik powinien zdobyć co najmniej 30 punktów na 60 możliwych (30 praktyka + 30 teoria).

Dzień 1 (8 godzin szkolenia)

- Praca w architekturze monorepo
- Tworzenie współdzielonej biblioteki
- Signals w architekturze Smart / Dumb components
- Tworzenie Component Provider i resource signal
- Profilowanie i poprawa wydajności aplikacji
- Tworzenie initializerów i injection tokenów

Dzień 2 (8 godzin szkolenia)

- Tworzenie dyrektyw strukturalnych i guardów
- Korzystanie z linked signal i deferred loading
- Tworzenie własnych kontrolek formularza
- Modyfikowanie formularza i walidacji
- Dynamiczne renderowanie komponentów
- Definiowanie routingu drugorzędneho i interceptorów

Dzień 3 (8 godzin szkolenia)

- Implementacja PWA w Angular
- Wysyłanie i odbieranie push notifications
- Zarządzanie stanem aplikacji z NgRx
- Testowanie E2E

- Stosowanie architektury Micro Frontends
- Test wiedzy (walidacja), podsumowanie i ewaluacja szkolenia przez uczestników

Harmonogram

Liczba przedmiotów/zajęć: 23

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
1 z 23 Setup aplikacji Angular (wykład i praktyka)	Tomasz Borowski	25-03-2026	09:00	10:00	01:00
2 z 23 Praca w architekturze monorepo (wykład i praktyka)	Tomasz Borowski	25-03-2026	10:00	11:00	01:00
3 z 23 Tworzenie współdzielonej biblioteki (wykład i praktyka)	Tomasz Borowski	25-03-2026	11:00	12:30	01:30
4 z 23 Przerwa	Tomasz Borowski	25-03-2026	12:30	13:00	00:30
5 z 23 Architektura Smart/Dumb components (wykład i praktyka)	Tomasz Borowski	25-03-2026	13:00	14:00	01:00
6 z 23 Tworzenie component providerów (wykład i praktyka)	Tomasz Borowski	25-03-2026	14:00	15:00	01:00
7 z 23 Profilowanie i poprawa wydajności aplikacji (wykład i praktyka)	Tomasz Borowski	25-03-2026	15:00	16:00	01:00
8 z 23 Tworzenie initializerów i injection tokenów (wykład i praktyka)	Tomasz Borowski	25-03-2026	16:00	17:00	01:00

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
9 z 23 Tworzenie dyrektyw strukturalnych (wykład i praktyka)	Tomasz Borowski	26-03-2026	09:00	10:00	01:00
10 z 23 Tworzenie operatorów RxJS (wykład i praktyka)	Tomasz Borowski	26-03-2026	10:00	11:00	01:00
11 z 23 Współdzielenie observables (wykład i praktyka)	Tomasz Borowski	26-03-2026	11:00	12:30	01:30
12 z 23 Przerwa	Tomasz Borowski	26-03-2026	12:30	13:00	00:30
13 z 23 Tworzenie własnych kontrolek formularza (wykład i praktyka)	Tomasz Borowski	26-03-2026	13:00	14:00	01:00
14 z 23 Modyfikowanie formularza i walidacji (wykład i praktyka)	Tomasz Borowski	26-03-2026	14:00	16:00	02:00
15 z 23 Dynamiczne renderowanie komponentów (wykład i praktyka)	Tomasz Borowski	26-03-2026	16:00	17:00	01:00
16 z 23 Implementacja interceptorów HTTP (wykład i praktyka)	Tomasz Borowski	27-03-2026	09:00	10:00	01:00
17 z 23 Definiowanie routingu drugorzędowego (wykład i praktyka)	Tomasz Borowski	27-03-2026	10:00	11:00	01:00

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
18 z 23 Implementacja PWA w Angular (wykład i praktyka)	Tomasz Borowski	27-03-2026	11:00	12:30	01:30
19 z 23 Przerwa	Tomasz Borowski	27-03-2026	12:30	13:00	00:30
20 z 23 Cachowanie danych w Service Worker (wykład i praktyka)	Tomasz Borowski	27-03-2026	13:00	14:00	01:00
21 z 23 Zarządzanie stanem aplikacji z NgRx (wykład i praktyka)	Tomasz Borowski	27-03-2026	14:00	15:30	01:30
22 z 23 Korzystanie z Angular Signals i Deferred Views (wykład i praktyka)	Tomasz Borowski	27-03-2026	15:30	16:30	01:00
23 z 23 Test wiedzy (walidacja)	-	27-03-2026	16:30	17:00	00:30

Cennik

Cennik

Rodzaj ceny	Cena
Koszt przypadający na 1 uczestnika brutto	3 196,77 PLN
Koszt przypadający na 1 uczestnika netto	2 599,00 PLN
Koszt osobogodziny brutto	133,20 PLN
Koszt osobogodziny netto	108,29 PLN

Prowadzący

Liczba prowadzących: 1



1 z 1

Tomasz Borowski

Programista aplikacji internetowych

Senior Front-end Developer pracujący na co dzień przy implementacji złożonych aplikacji biznesowych w frameworku Angular.

magister - Uniwersytet Jagielloński

inżynier - BWS im. Tyszkiewicza

Tomasz był wielokrotnie speakerem na konferencjach lokalnych oraz międzynarodowych - w tym Agile by Example, Agile Lean Europe, Agile Cambridge. Od 2017 roku poprowadził ponad 130 szkoleń w technologii Angular

Informacje dodatkowe

Informacje o materiałach dla uczestników usługi

Uczestnicy szkolenia otrzymują dostęp do platformy szkoleniowej, na której znajdują się wszystkie niezbędne materiały szkoleniowe:

- omówienia tematów szkoleniowych
- przykłady zastosowań (fragmenty kodu)
- zadania szkoleniowe

Warunki techniczne

Szkolenie prowadzone jest w formie zdalnej za pośrednictwem serwisów

<https://clickmeeting.com/>

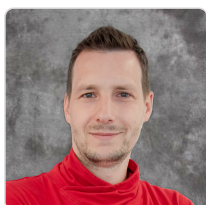
oraz

<https://whereby.com>

Uczestnik szkolenia powinien spełnić następujące warunki techniczne:

- posiadać własny komputer w specyfikacji sprzętowej umożliwiającej uruchomienie przeglądarki internetowej (zalecamy Chrome) oraz edytora kodu (zalecamy Webstorm lub Visual Studio Code)
- posiadać dostęp do stabilnego łącza internetowego, umożliwiającego audiowizualną komunikację w czasie rzeczywistym
- posiadać zainstalowane środowisko NodeJS w wersji co najmniej 20.19.x

Kontakt



Tomasz Borowski

E-mail info.tbprojects@gmail.com

Telefon (+48) 517 451 608