

## SZKOLENIE ŚREDNIO ZAAWANSOWANE

---

# Tworzenie aplikacji z użyciem ReactJS

## REACT

**Czas trwania: 3 dni**

Jedna z najpopularniejszych bibliotek JavaScript

### Cele szkolenia

---

- Zapoznanie uczestników z założeniami, na których oparta jest biblioteka React oraz architektura Redux
- Zdobycie praktycznych umiejętności związanych z wykorzystaniem tych narzędzi do budowy wydajnych aplikacji webowych

### Zalety

---

- Szkolenie ReactJS - tworzenie aplikacji webowych w maksymalnym stopniu nastawione jest na aspekt praktyczny
- Jego głównym założeniem jest rozwinięta część warsztatowa obejmująca zarówno budowę niezależnego projektu jak i stworzenie pojedynczych komponentów możliwych do wykorzystania w ramach projektów opartych o inne frameworki
- Praktyka przed teorią - wszystkie szkolenia technologiczne prowadzone są w formie warsztatowej. Konieczna teoria jest wyjaśniana na przykładzie praktycznych zadań
- Konkretność umiejętności - w ramach każdego szkolenia rozwijamy praktyczne umiejętności związane z daną technologią i tematyką
- Nauka z praktykami - wszyscy trenerzy na co dzień pracują w projektach, gwarantuje to dostęp do eksperckiej wiedzy i praktycznego know-how

### Dla kogo?

---

- Szkolenie ReactJS - tworzenie aplikacji webowych przeznaczone jest dla programistów posiadających podstawową wiedzę w zakresie HTML i JavaScript, którzy chcą poszerzyć swoje kompetencje związane z tworzeniem wydajnych aplikacji działających po stronie przeglądarki

### Wymagania

---

- Rozeznanie w podstawach programowania w JavaScript lub innym języku skryptowym



- Podstawowa wiedza dotycząca języka HTML i CSS



## Program

---

1. Wprowadzenie
  - a. Czym jest React i jakie problemy rozwiązuje?
  - b. Omówienie ekosystemu i narzędzi
2. Struktury JavaScript w aplikacjach React
  - a. Programowanie funkcyjne z ECMAScript 5
  - b. Wstęp do ECMAScript 2015+
    - Przydatne konstrukcje (fat arrows, destructuring, rest parameters, ...)
    - Transpilacja nowych wersji ECMAScript
3. Narzędzia
  - a. Praca z npm
  - b. Transpilacja kodu (Babel)
  - c. Webpack i Webpack Dev Server
4. Podstawy React
  - a. Idea Virtual DOM
  - b. JSX
    - Składnia JSX
    - Atrybuty HTML a właściwości komponentów
    - React w JS vs React w JSX
  - c. Komponenty React
    - Komponenty funkcyjne i klasowe
    - Tworzenie i renderowanie komponentu
    - Dane w komponencie: props i state
    - Praktyczne różnice między właściwościami a stanem
    - Walidacja właściwości - PropTypes
    - Przekazywania właściwości kolejnym komponentom
    - Obsługa zdarzeń
    - Komponenty prezentacyjne, kontrolne i HOC
  - d. Komponenty klasowe
    - Cykl życia komponentu
    - Praktyczne zastosowania poszczególnych etapów cyklu życia
  - e. Komponenty funkcyjne
    - Prowadzenie do hooks
    - Zastosowania poszczególnych wbudowanych hooków
    - Tworzenie własnych hooków
  - f. Formularze
    - Kontrolowane komponenty
    - Uzyskiwanie referencji do komponentu
    - Uzyskiwanie dostępu do obiektu DOM
    - Praca z Formik
  - g. Stylowanie
    - Przegląd dostępnych rozwiązań



- CSS Modules

- styled-components

- h. Optymalizacja aplikacji dla produkcji za pomocą Webpack

- i. Routing

  - Statyczne ścieżki

  - Dynamiczne ścieżki

  - Zagnieżdżenia

  - Komponent <Link/>

  - Obsługa błędów

- j. Dobre praktyki

- 5. Redux - zarządzanie stanem aplikacji

  - a. Omówienie architektury

  - b. Modelowanie stanu

  - c. Reducers

  - d. Łączenie React z Redux

  - e. Actions i action creators

  - f. Metody implementacji niemutowalnych zmian

  - g. Kiedy Redux a kiedy Context?

  - h. Praca z Redux Dev Tools

- 6. MobX

  - a. Omówienie architektury

  - b. Praktyczne zastosowania

  - c. Porównanie z Redux

- 7. Testowanie

  - a. Wprowadzenie do TDD w JavaScript

  - b. Struktura testów jednostkowych

  - c. Praca z Jest

  - d. Możliwości React Testing Library

