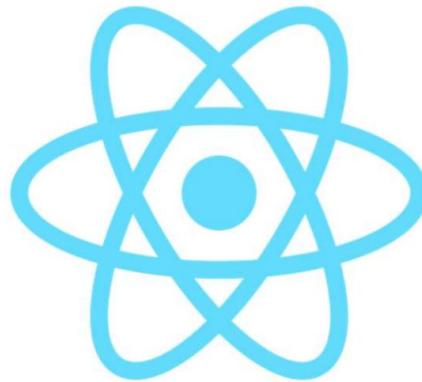


## Opis szkolenia

# React Native



### Zapraszamy na 3-dniowe warsztaty szkoleniowe z React Native.

Podczas szkolenie zaimplementujesz własną aplikację, która od pierwszych linii kodu będzie wspierała jednocześnie Androida i iOS. Zobaczysz, jak dostosować elementy aplikacji do wykorzystania elementów dostępnych tylko w jednym z tych systemów. Dowiesz się czym się różni React Native od w pełni natywnych aplikacji jak i od tych opartych o Ionic/Cordovę.

### Dlaczego React Native?

React jest wydajną biblioteką do obsługi interfejsu użytkownika. Dzięki wykorzystaniu koncepcji komponentów ma również proste API oraz pozwala na względnie proste użycie poza pierwotnym środowiskiem przeglądarki. Inżynierowie Facebooka wykorzystali ten fakt do przygotowania zestawu API oraz komponentów React'a, które nie dość, że są dostępne bezpośrednio w JavaScript, to są renderowane jako natywne kontrolki mobilnego systemu operacyjnego.

Dzięki temu programiści iOS i Android mogą współdzielić najważniejsze części aplikacji wciąż pisząc aplikacje natywne, a programiści JavaScript mogą pisać aplikacje natywne korzystając z języka i bibliotek, które doskonale znają.

Dodając fakt bardzo silnej społeczności stojącej zarówno za React Native jak i samym Reactem wydaje się, że React Native jest jedną z najlepszych opcji to stworzenia aplikacji mobilnej.

## Dla kogo?

Szkolenie i przekazywana na nim wiedza jest kierowana do osób programujących w JavaScript.

## Wymagana wiedza

Znajomość JavaScript co najmniej w specyfikacji EcmaScript 2015 oraz znajomość biblioteki React.

## Format

Szkolenie przewidziane jest na 3 dni, podczas których uczestnicy utworzą od zera własną aplikację - notatnik z obsługą synchronizacji notatek z serwerem oraz możliwością zapisywania grafik i miejsc na mapie. Aplikacja będzie zgodna z platformami Android oraz iOS. Zgodnie z zasadą maksimum praktyki - największy nacisk będzie położony na pracę z kodem.

# Agenda

1. Przygotowanie nowego projektu. Struktura projektu React Native. Uruchamianie i debugowanie aplikacji.
2. Podstawowe komponenty i style. Zarządzanie pozycją elementów. Strategie zarządzania stylami.
  - a. komponenty: View, Text, Image
  - b. model flexbox
  - c. współdzielenie stylów przez komponenty i moduły
3. Obsługa list: wyświetlanie danych oraz wydajność różnych rozwiązań.
  - a. FlatList
  - b. SectionList
  - c. ScrollView
4. Obsługa interakcji z wykorzystaniem natywnych oraz własnych kontrolki. Dostosowanie widoku do wyświetlenia klawiatury
  - a. Button
  - b. TouchableHighlight
  - c. TouchableNativeFeedback
  - d. TouchableOpacity
  - e. TouchableWithoutFeedback
  - f. KeyboardAvoidingView
5. Obsługa wielu widoków z biblioteką react-navigation
  - a. createStackNavigator
  - b. createTabNavigator

- c. createDrawerNavigator
  - d. właściwość navigation
  - e. widoki: StackView (StackViewCard, Header), SwitchView, Tabs, Drawer
6. Korzystanie z dostępnych API
- a. CameraRoll
  - b. AsyncStorage
  - c. fetch
  - e. Share
7. Dostosowanie elementów aplikacji pod konkretną platformę.
- a. sprawdzanie systemu operacyjnego oraz jego wersji
  - b. tworzenie modułów specyficznych dla danej platformy
8. Korzystanie z zewnętrznych bibliotek korzystających z natywnych API na przykładzie react-native-maps
9. Przygotowanie aplikacji do wydania
- a. budowanie aplikacji
  - b. konfiguracja aplikacji do wydania w serwisach Google Play i AppStore
  - c. korzystanie z serwisu CodePush w celu zautomatyzowania procesu budowania
  - d. korzystanie z serwisu Fastlane w celu zautomatyzowania procesu budowania
10. Testowanie
- a. Pisanie testów jednostkowych z wykorzystaniem narzędzia Jest