

Kod szkolenia: **ANDROID/ASYNC**

Tytuł szkolenia: **Asynchroniczność na platformie Android (Kotlin)**

Dni: 4

## Opis:

### Cele szkolenia:

- Wprowadzenie do programowania reaktywnego oraz użycia współprogramów
- Zapoznanie z RxJava oraz Coroutines
- Nabycie wiedzy / umiejętności umożliwiających samodzielne wykorzystanie RxJava oraz Coroutines w projekcie

### Mocne strony szkolenia:

- Szkolenie w maksymalnym stopniu nastawione jest na aspekt praktyczny
- Jego głównym elementem jest rozwinięta część warsztatowa obejmująca zarówno budowę złożonego projektu jak i niezależne ćwiczenia pomagające zrozumieć i utrwalić poruszane zagadnienia
- Uczestnicy poznają dobre praktyki oraz sposoby rozwiązywania typowych problemów, które pojawiają się przy pisaniu kodu asynchronicznego

### Adresaci:

- Szkolenie przeznaczone dla osób już potrafiących tworzyć aplikacje na platformę Android

### Wymagania:

- Od uczestników szkolenia wymagana jest praktyczna umiejętność programowania w języku Kotlin lub Java
- Od uczestników szkolenia wymagana jest podstawowa platformy Android oraz doświadczenie w tworzeniu na nią aplikacji

### Program szkolenia:

- Wstęp
  - Kod asynchroniczny
  - Wielowątkowość w Androidzie
  - Handlers, Threads, Callbacks



- Programowanie reaktywnego
  - Idea programowania reaktywnego
  - RxJava - Strumienie, operatory i zarządzanie wątkami
- Observable - obserwowalne strumienie danych
  - Tworzenie, subskrypcja, zwalnianie zasobów
  - Zdarzenia onNext, onError, onComplete
  - Hot vs. Cold Observable
  - Single
  - Subject i jego implementacje
- Operatory
  - Tworzenie strumieni
  - Transformacja
  - Filtrowanie
  - Kombinowanie
  - Obsługa błędów
  - Przetwarzanie warunkowe
  - Agregacja
  - Backpressure
  - Inne
- RxAndroid w praktyce
  - Architektura MVVM
  - Reaktywny interfejs użytkownika
  - Komponowanie strumieni
  - Przełączanie wątków
  - Dobór operatorów
- Coroutines
  - Subroutines i coroutines
  - Różnica między kodem suspending a blocking
  - Tworzenie coroutines z CoroutineBuilder
  - CoroutineContext oraz CoroutineScope
  - Dispatchers - praca z wątkami
  - Channels
  - Flows
  - Porównanie do RxJava
- Adaptacja istniejącego kodu asynchronicznego
- Typowe problemy
- Testowanie kodu asynchronicznego
  - Idling resource
  - LiveData
  - RxJava Plugins
  - Coroutines

