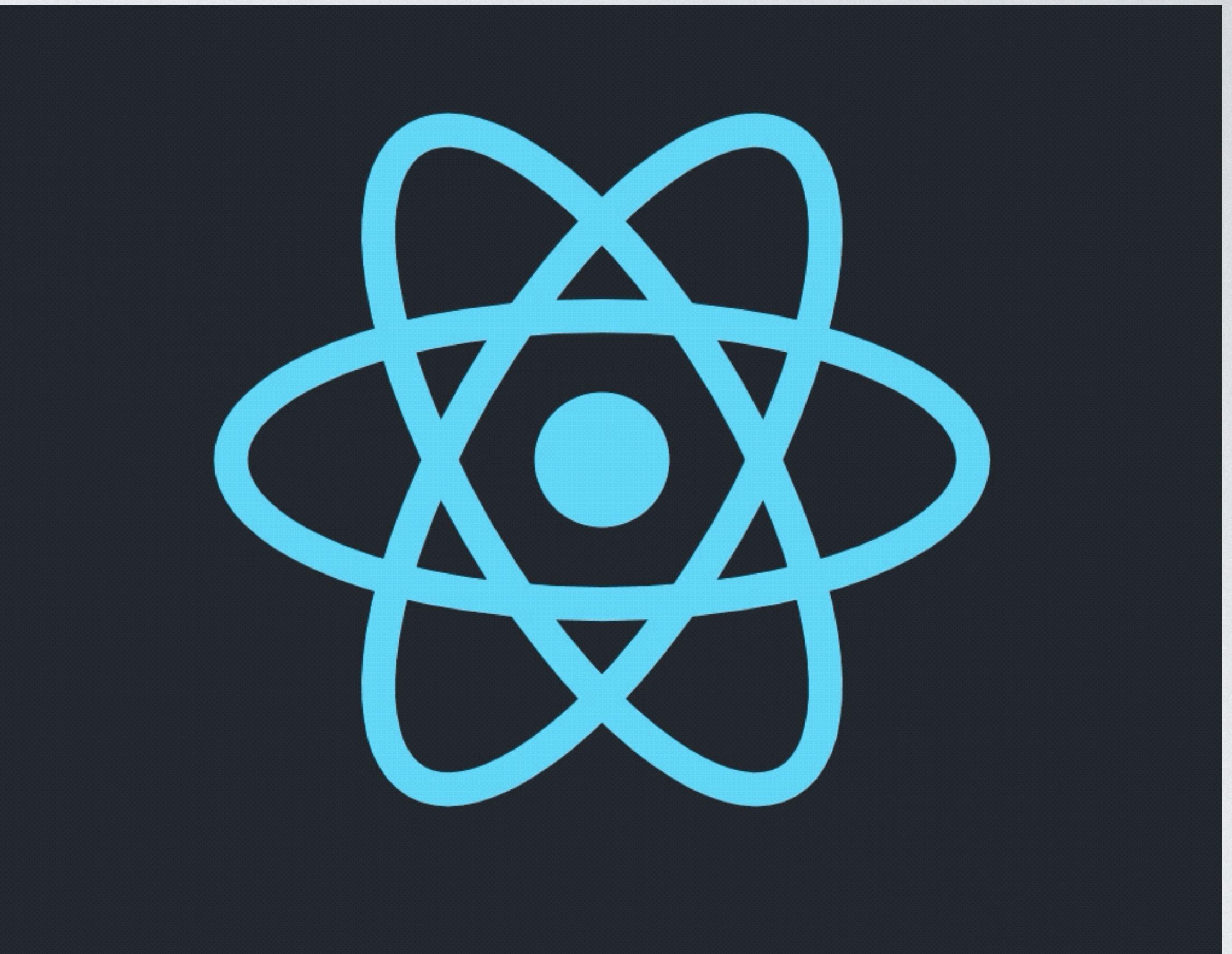


REACT NATIVE



OMNIE



- Front End developer w Leocode



- Koduję fronty od prawie 7 lat



- Wiking



APLIKACJE MOBILNE

Apple

Swift

App Store

Android

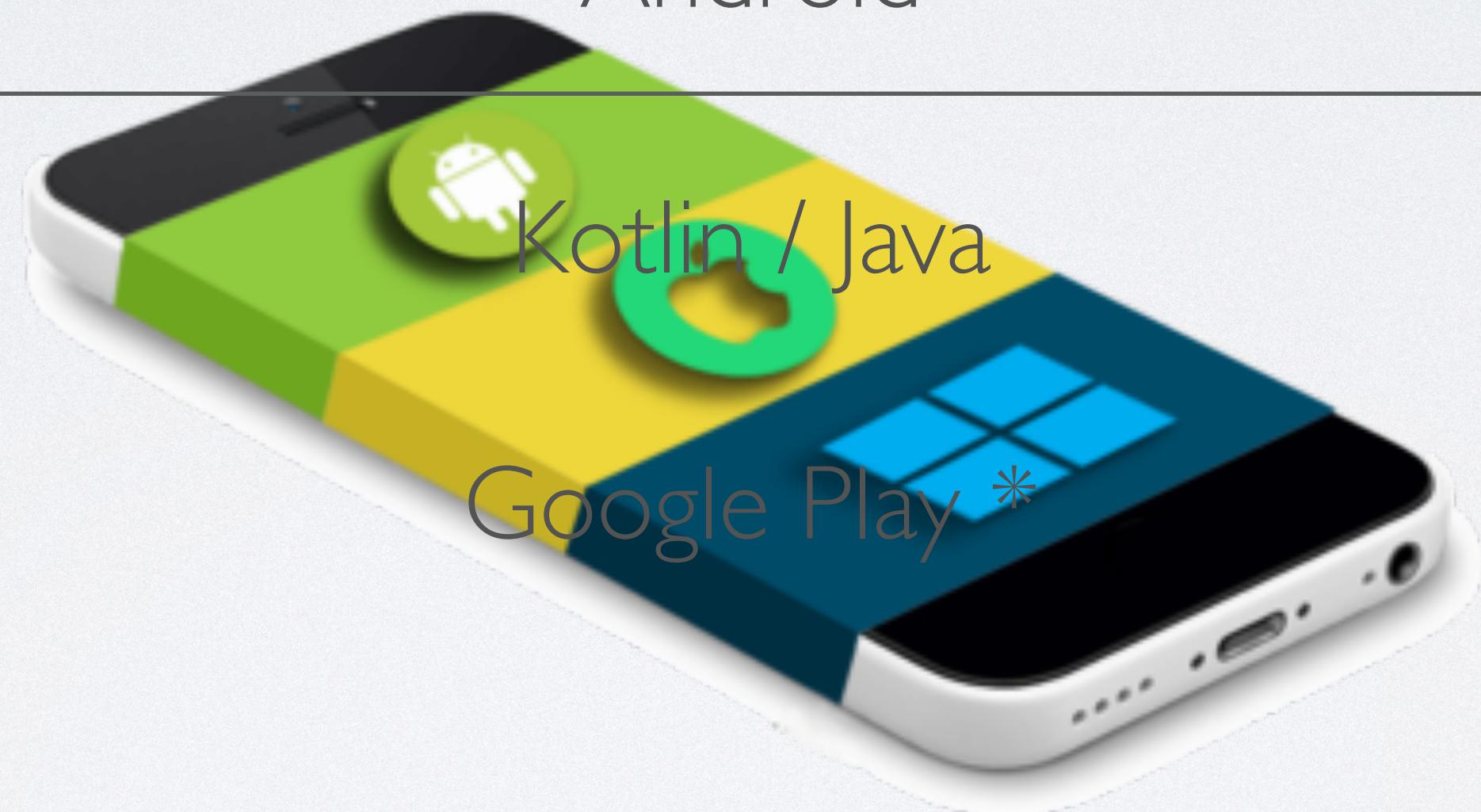
Kotlin / Java

Google Play *

Windows

C#

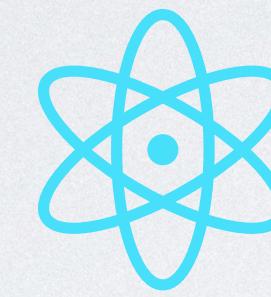
Dead



Hybrid

APLIKACJE MOBILNE - HYBRID

QT



ReactNative

Ionic

Flutter

NativeScript

Xamarin

PhoneGap

VueForia

C#

C++

JavaScript

AngularJS

VueJS

ReactJS

Dart

PWA

Progressive Web Application

Will be dead at 2020

WHY

React Native

Learn once, write anywhere.

ΓΕΩΡΓΙΟΝ ΟΥΝΕΙ ΜΗΣ ΣΥΛΛΟΓΕΙ.

WHAT

- Framework - nie biblioteka!
- Nadal w becie*
- Dodatek do aplikacji natywnych lub pełnoprawna aplikacja

HOW

- Kod w JavaScript, kompliuje się do aplikacji natywnej*
- Prosty cross platform
- Hot Reload jak w React!

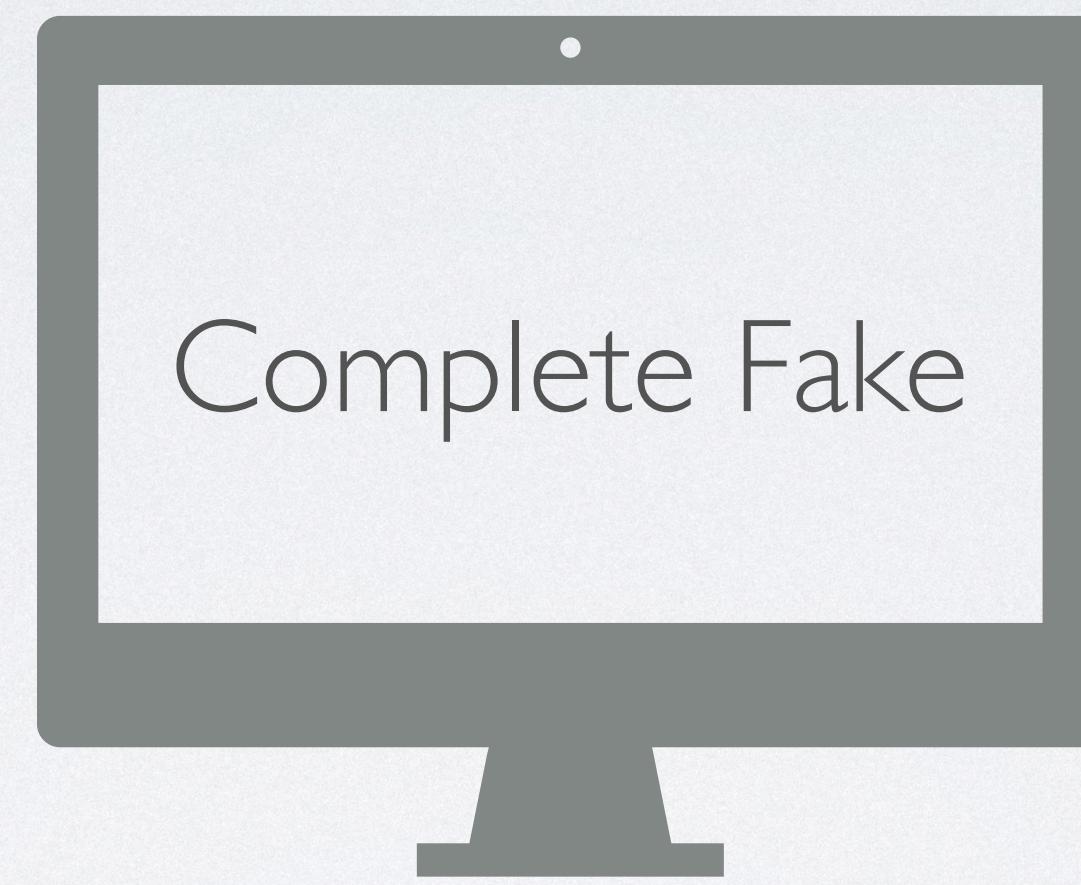
TROCHĘ GORZKIEJ PRAWDY

- iOS first
- Windows z problemami
- Android Studio jest super zasobne
- Xcode nie jest JS deweloper friendly

GDZIE



Emulator



Symulator*



Prawdziwe urządzenie

WYMAGANIA

- Komputer z HyperV (nie dotyczy Apple)
- Windows PRO+, macOS (*lub Hackintosh*), Linux
- Telefon z Androidem 5+* lub iOSem 10+

WYMIAGANIA V2

Android

- Android Studio
- Watchman
- JavaDevelopmentKit 8
- Chocolatey (Windows)
- Python2 (Windows)

iOS

- Xcode
- CocoaPods
- Homebrew
- Watchman

APLIKACJA

Debug

Release

EXPO

The fastest way from an idea to a native experience

Expo is a free and complete workflow for creating next-generation
apps for iOS and Android

expo.dev

EXPO - FAQ

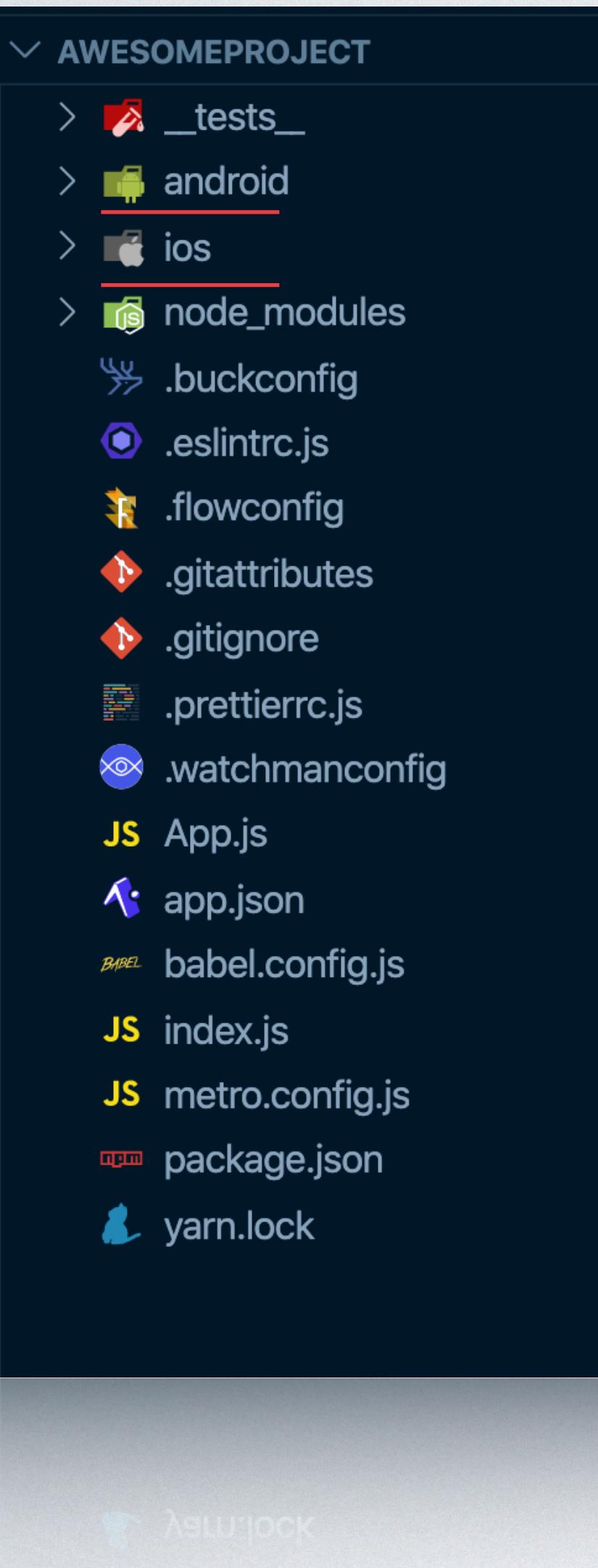
- Czy będziemy używać go na zajęciach? **Nie**
- Dlaczego Expo to nie najlepszy wybór na zajęcia? **Nie pozwala, na instalacje natywnych bibliotek, ogranicza możliwości uczenia się**
- Czy Expo ułatwia pracę z React Native w prawdziwym świecie? **Tak i Nie**
- Bardzo chcę używać Expo, czy mogę? **Nie na zajęciach, poza - śmiałko**
- A jeśli u mnie nie działa React Native bo... **To użyj Expo, jak zadziała, to korzystaj. Trudno...**

ZACZYNAMY

```
npx react-native  
init AwesomeProject
```

COLLEGE OF MASTERS IN JOURNALISM

APLIKACJA



- **Buck** - optymalizacja buildów *by Facebook*
- **Flow** - typowanie aplikacji - *by Facebook*
- **Prettier** - upiększa kod według zasad
- **metro** - builder (aka webpack) *by Facebook*

ANDROID & IOS

- Katalog android & ios to odpowiednio katalogi projektów **natywnych**
- **Będziemy** edytować ich zawartość
- Te katalogi otwieramy przy pomocy **AS lub Xcode**

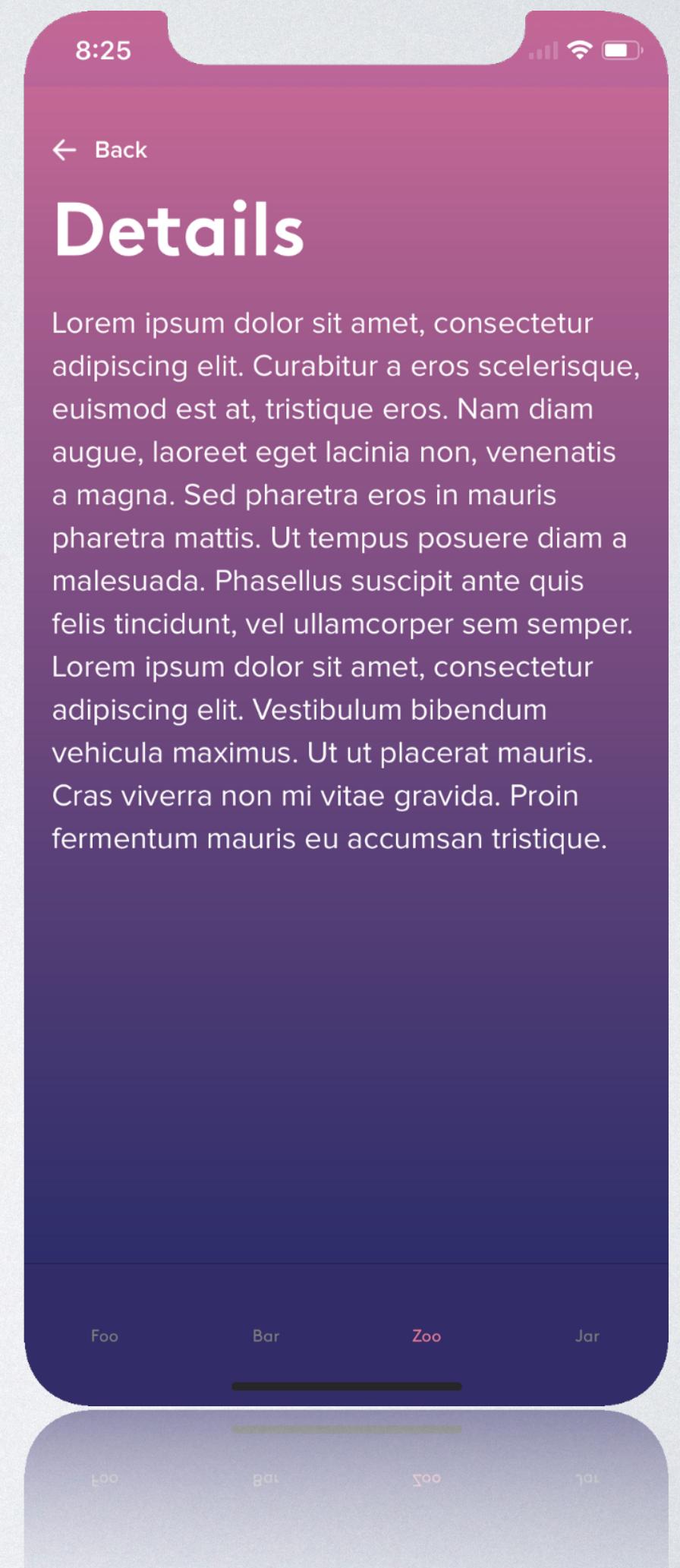
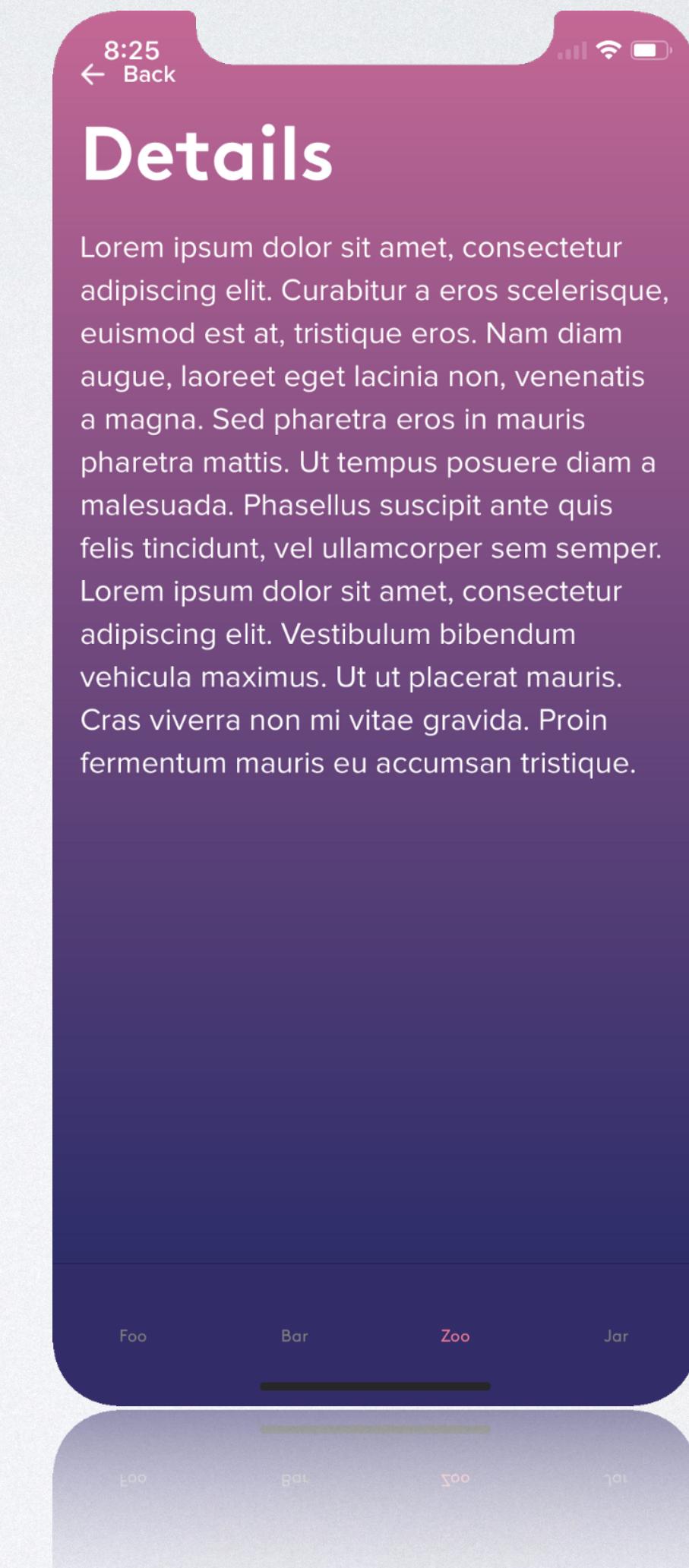
APP.JS

WYKŁAD 2

- Komponenty
- Rzecz o stylach
- Dokumentacja
- Architektura React Native
- Instalowanie modułów
- AsyncStorage

KOMPONENTY - VIEW

- Bazowy kontener
- Jest zamieniany na odpowiadający mu natywny element: UIView, <div>, android.view
- Częściwo obsługuje eventy związane z dotykiem



KOMPONENTY - SCROLLVIEW

- Podstawowy komponent wyświetlający dzieci
- Pozwala na przesuwanie palcem elementów na ekranie

```
<ScrollView style={{flex: 1}}>
  <View style={{height: 10, backgroundColor: 'red'}}></View>
  <View style={{height: 10, backgroundColor: 'green'}}></View>
  <View style={{height: 10, backgroundColor: 'blue'}}></View>
  <View style={{height: 10, backgroundColor: 'yellow'}}></View>
  <View style={{height: 10, backgroundColor: 'red'}}></View>
  <View style={{height: 10, backgroundColor: 'green'}}></View>
  <View style={{height: 10, backgroundColor: 'blue'}}></View>
  <View style={{height: 10, backgroundColor: 'yellow'}}></View>
  <View style={{height: 10, backgroundColor: 'red'}}></View>
  <View style={{height: 10, backgroundColor: 'green'}}></View>
  <View style={{height: 10, backgroundColor: 'blue'}}></View>
  <View style={{height: 10, backgroundColor: 'yellow'}}></View>
  <View style={{height: 10, backgroundColor: 'red'}}></View>
  <View style={{height: 10, backgroundColor: 'green'}}></View>
  <View style={{height: 10, backgroundColor: 'blue'}}></View>
</ScrollView>;
```

KOMPONENTY - TEXT

- Każdy string **musi** być w tagu
`<Text>`

`x <View>Header</View>`

`✓ <Text>Header</Text>`

KOMPONENTY - TEXT

- <Text> też jest kontenerem

- Flexbox nie działa wewnątrz

<Text>

```
<Text>
  <Text>This is Header</Text>
  <Text>This is subHeader</Text>
</Text>

// This is HeaderThis is subHeader
```

KOMPONENTY - TEXT

- <Text> też jest kontenerem
- Flexbox nie działa wewnątrz

<Text>

```
<View>
  <Text>This is Header</Text>
  <Text>This is subHeader</Text>
</View>

// This is Header
// This is subHeader
```

KOMPONENTY - TEXT

- Text możemy stylować wewnętrz innego Textu

```
<Text>This is <Text style={{color: 'red'}}>Header</Text></Text>
```

KOMPONENTY - TOUCHABLE

- Button to inny komponent - nie modyfikowalny za mocno
- 4 rodzaje elementów, posiadających obsługę dotyku
- 1 Child required

KOMPONENTY - TOUCHABLE

- **NativeFeedback**
- Tylko Android
- Wykorzystuje natywne zachowanie touch w UI Androida

KOMPONENTY - TOUCHABLE

- **Highlight**
- iOS / Android
- Modyfikuje jasność i przezroczystość elementu

KOMPONENTY - TOUCHABLE

- **Opacity**
- iOS / Android
- Modyfikuje jasność i przeźroczość elementu

KOMPONENTY - TOUCHABLE

- **WithoutFeedback**
- iOS / Android
- Nie dostarcza żadnej wizualnej odpowiedzi na dotyk

KOMPONENTY - TOUCHABLE

- **Po co aż 4?**
- Opacity* - używamy na iOS do przycisków nie posiadających tła
- Highlight - używamy na iOS do przysieków o wyraźnych konturach i kolorach
- NativeFeedback - używamy na Android
- WithoutFeedback - nie używamy, chyba, że mamy jakiś super powód

KOMPONENTY - IMAGE

- Android: nie obsługuje gifów*
- Nie wspiera SVG
- Działa jak zwykłe
- Należy ustalać wielkość obrazka!



KOMPONENTY - TEXTINPUT

- Fundamentalny komponent
- Pozwala modyfikować klawiaturę
zależnie od platformy
- Jest bardzo ciężko stylowalny

```
<TextInput onChangeText={(text) => setText(text)} />
```

KOMPONENTY - FLATLIST

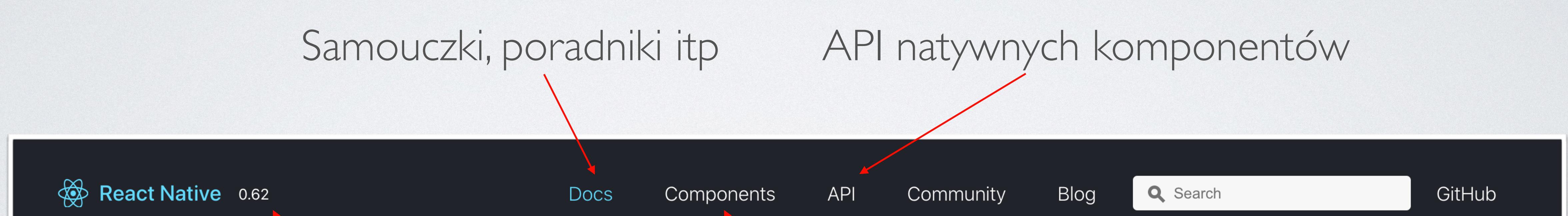
- Tworzy listę elementów
- Obsługuje eventy na krawędziach ekranu
- Lazy loading

```
<FlatList  
  data={[  
    {key: 1, name: Test}.  
    {key: 2, name: Test2}.  
    {key: 3, name: Test3}.  
  ]}  
  renderItem={JSX COMPONENT}  
>
```

STYLING

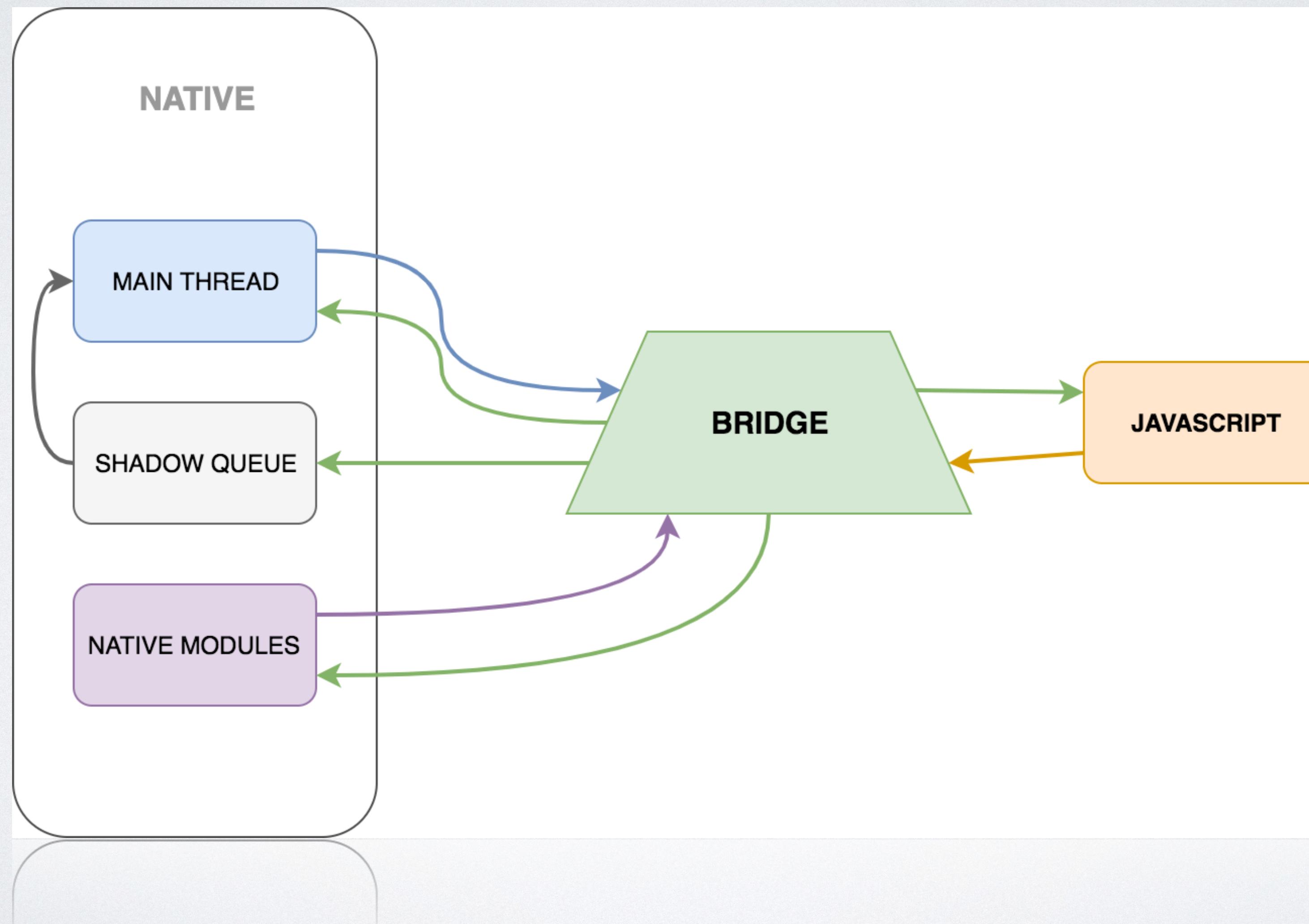
- StyleSheet !== plain Object
- camelCase
- Flexbox
- Brak dziedziczenia stylu
- Nie ma pixeli - są jednostki
- Aplikacja nie zna wymiarów urządzenia

DOKUMENTACJA

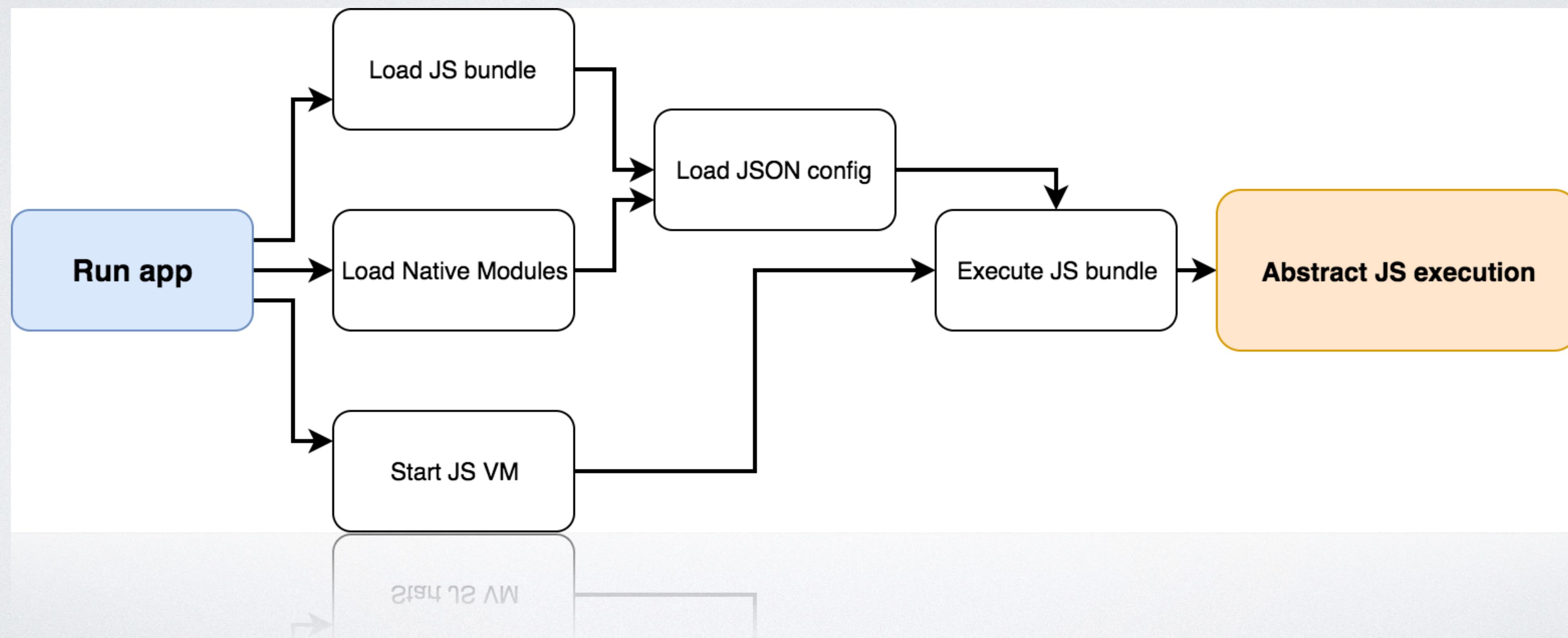


Wersja
zawsze należy wybrać wersję identyczną,
z tą w projekcie!

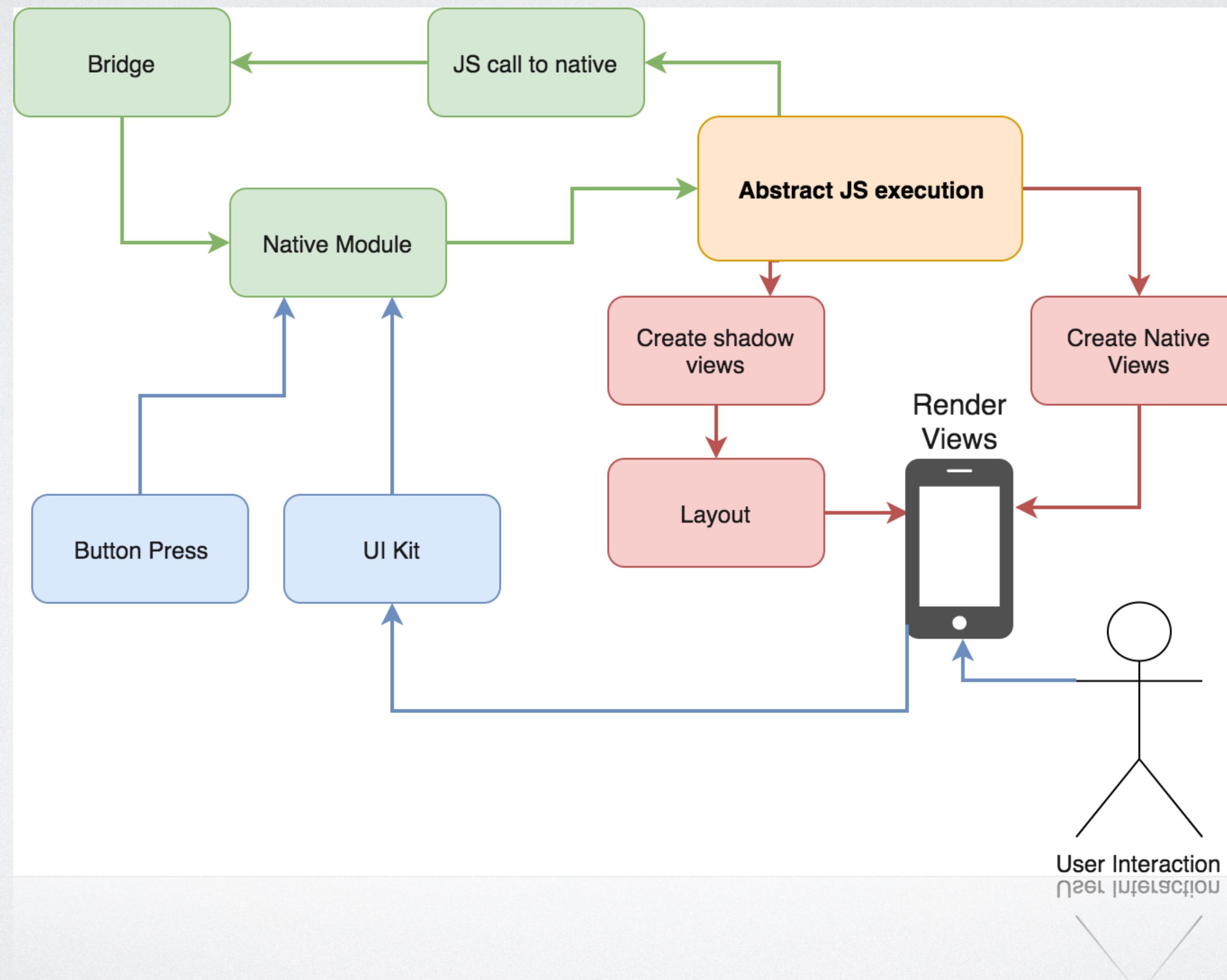
ARCHITEKTURA - SCHEMAT



ARCHITEKTURA - COMPILE



ARCHITEKTURA - RUNTIME



INSTALOWANIE NATYWNYCH KOMPONENTÓW

- *yarn add react-native-camera*
- Node modules: paczka zawierająca kod JS + kod Native
- **linkowanie**
- Clear build folder
- Rebuild app

ASYNCSTORAGE

- Nie zaszyfrowany
- Asynchroniczny
- Trwały
- Klucz-wartość

ASYNCSTORAGE

- Stosuje się nie tak jak locastorage w webie
- Przechowuje tylko String
- **Obecnie z zewnętrznej paczki!**