





Od ostatniego spotkania...

- 29 marca wydaliśmy Expo SDK^26
- Zmiany związane z kodami QR na iOS
- React Native v0.54, React v16.3.0 - zmiany w cyklu życia komponentów
- Pełna lista zmian na blogu: go.gl/kf518S



React v16.3.0

- Porzuca wsparcie dla metod:
componentWillMount, componentWillReceiveProps, componentWillUpdate
- Wprowadza nowe metody:
getSnapshotBeforeUpdate, getDerivedStateFromProps
- Context API
- createRef API
- Pełna lista zmian: goo.gl/rH27rN



Layout models

Cascading Style Sheets (CSS)

React Native Stylesheets

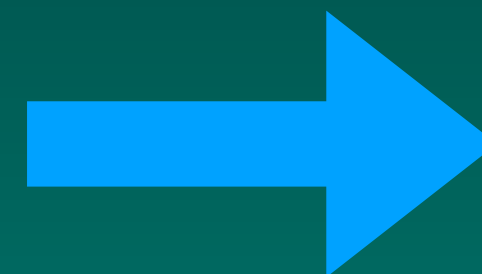
Block

Inline-block

Table

Flexbox

Grid



Flexbox



Flexbox Layout





Flex container rules

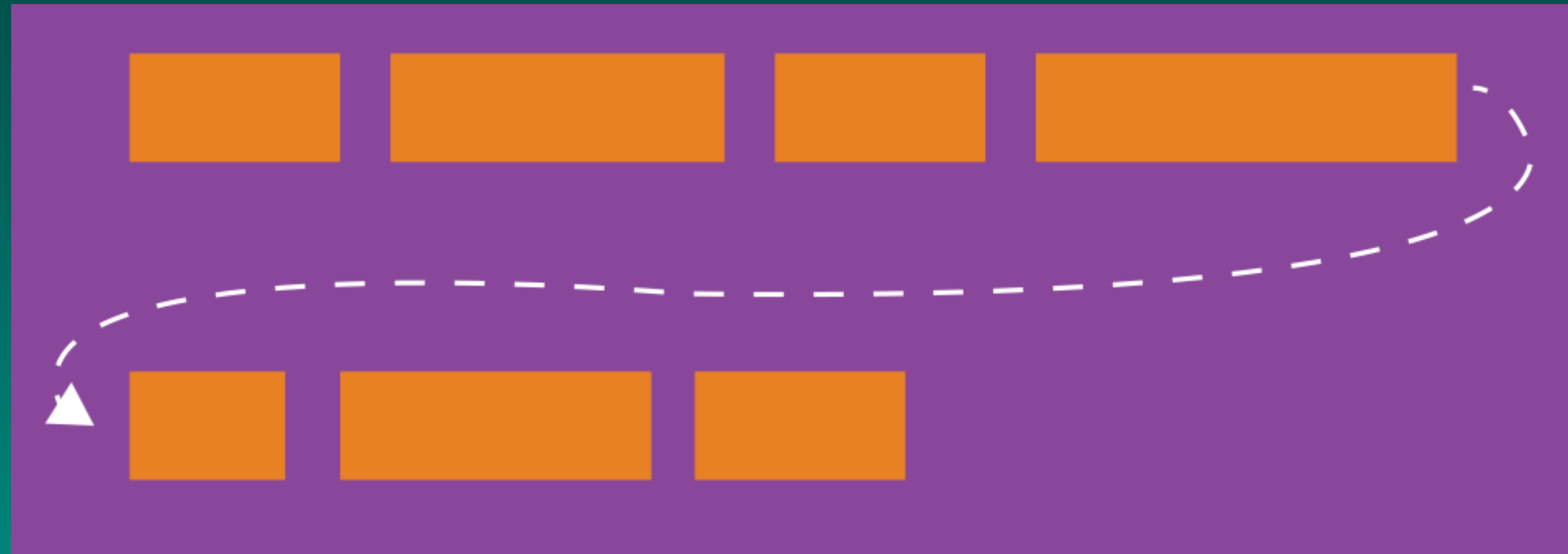
`flexDirection: 'row' | 'row-reverse' | 'column' | 'column-reverse'`





Flex container rules

flexWrap: *'wrap'* | *'nowrap'*





Flex container rules

justifyContent: *'flex-start'* | *'flex-end'* | *'center'* |
'space-between' | *'space-around'* | *'space-evenly'*

flex-start



flex-end



center



space-between



space-around



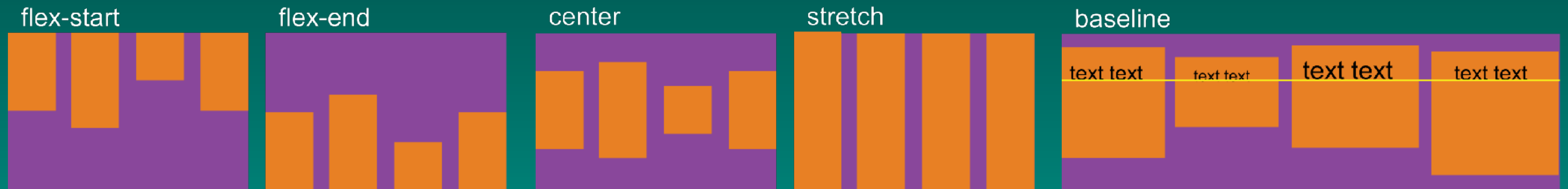
space-evenly





Flex container rules

alignItems: *'flex-start'* | *'flex-end'* | *'center'* | *'stretch'* | *'baseline'*



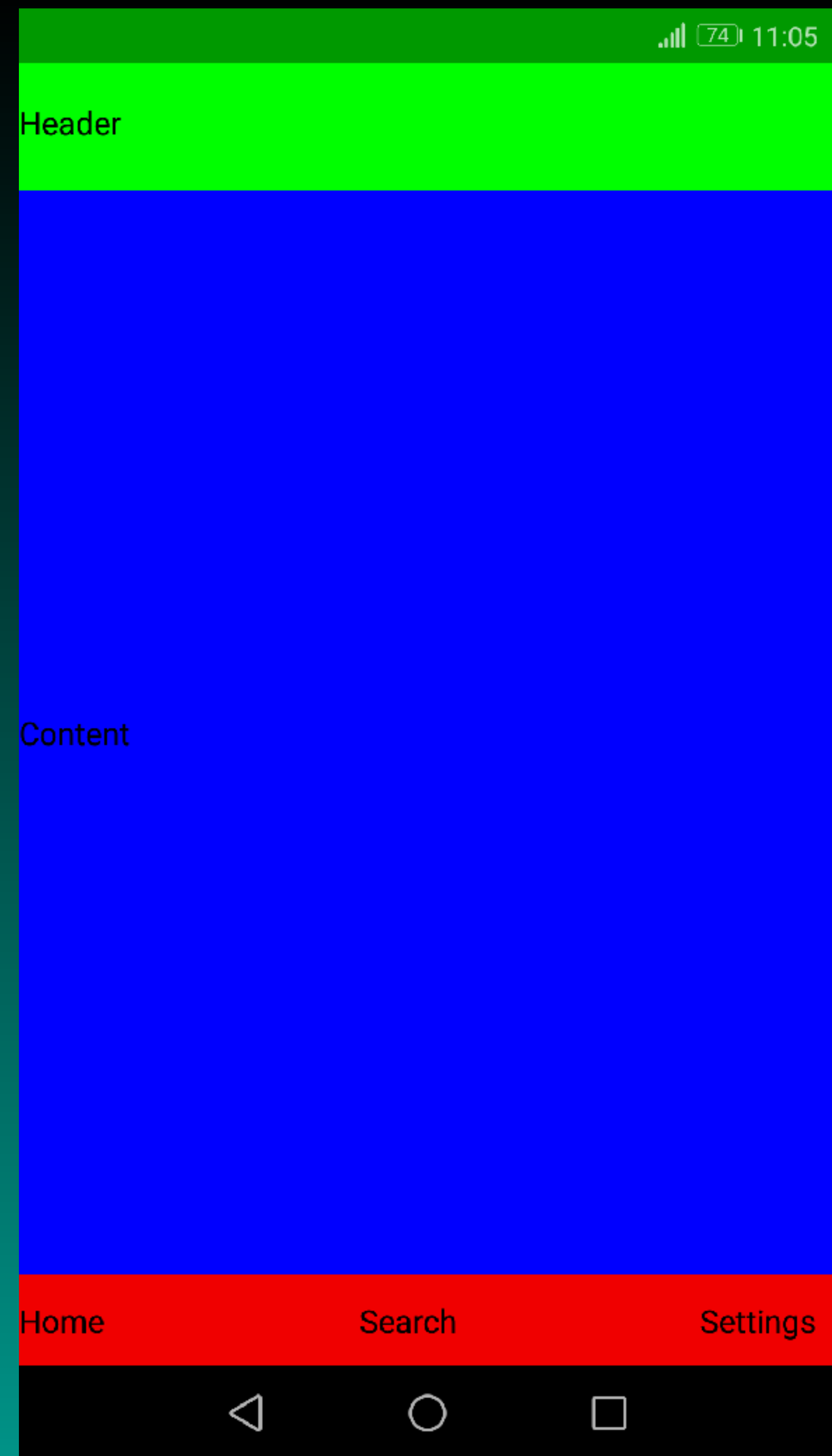
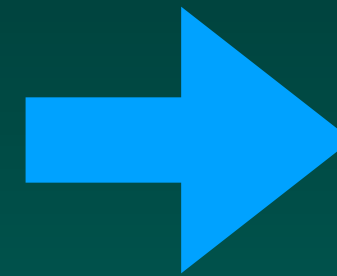


Flex item rule

flex: number

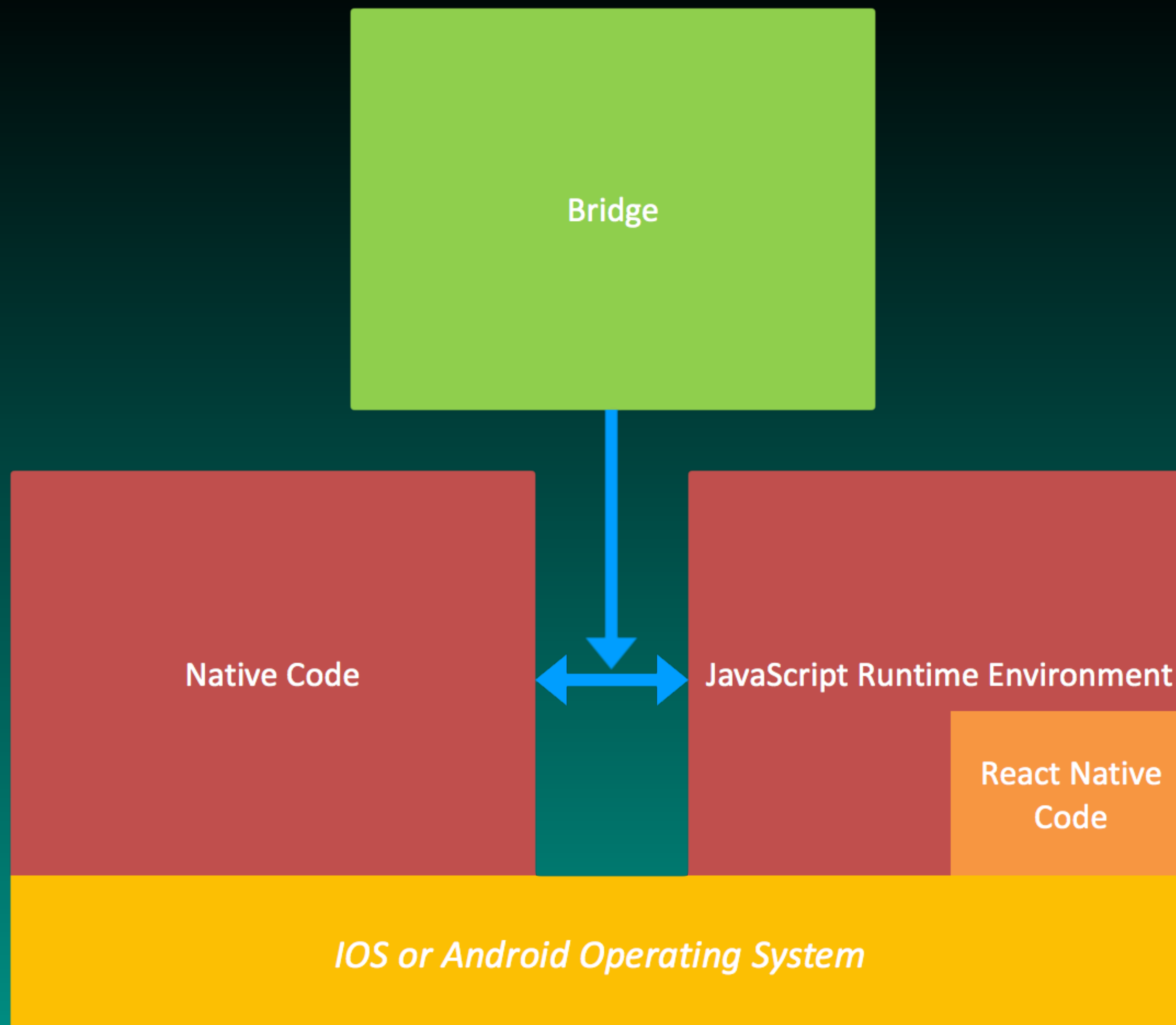



```
export default class App extends React.Component {
  render() {
    return (
      <View style={styles.container}>
        <View style={styles.header}>
          <Text>Header</Text>
        </View>
        <View style={styles.content}>
          <Text>Content</Text>
        </View>
        <View style={styles.footer}>
          <View style={styles.footerItem}>
            <Text>Settings</Text>
          </View>
          <View style={styles.footerItem}>
            <Text>Search</Text>
          </View>
          <View style={styles.footerItem}>
            <Text>Home</Text>
          </View>
        </View>
      </View>
    );
  }
}
```





Ale jak to działa?





MapView

- Komponent wyświetlający mapę
- Apple Maps lub Google Maps na iOS
- Google Maps na Androidzie
- Kilka dodatkowych komponentów i dokumentacja github.com/react-community/react-native-maps

```
import React from 'react';
import { MapView } from 'expo';

class App extends React.Component {
  render() {
    return (
      <MapView
        region={{
          latitude: 50.0613148,
          longitude: 19.9362257,
          latitudeDelta: 0.0922,
          longitudeDelta: 0.0421,
        }} />
    );
  }
}

export default App;
```




Location

- Pobieranie aktualnej pozycji GPS
- Nasłuchiwanie na zmiany pozycji
- Geocoding - pobieranie lokacji na podstawie adresu
- Reverse Geocoding - adres na podstawie lokacji
- Określenie kierunku magnetycznego

```
const location = await Location.getCurrentPositionAsync();
```



Font

- Ładowanie zewnętrznych czcionek
- Użycie poprzez regułę *'fontFamily'* w stylach
- Przykład na snacku snack.expo.io/rJhh-A3jM

```
import ComicSans from './assets/fonts/ComicSans.ttf';  
...  
await Font.loadAsync({ ComicSans });  
this.setState({ fontsLoaded: true });  
...  
<Text style={{ fontFamily: 'ComicSans' }}>Tekst</Text>
```




Inne

- Analytics (Amplitude, Segment)
- Audio/Video
- Calendar
- Camera
- Fingerprint
- GestureHandler
- GLView (WebGL 2.0)
- ImageManipulator
- ImagePicker
- Notifications
- Payments
- Permissions
- SecureStore
- SQLite



Zadanie

- Aplikacje, która potrafi wykryć kod QR i odczytać go na głos
- Moduły z Expo do użycia: BarCodeScanner, Speech, Permissions
- Do testów można użyć generatora kodów QR



Kolejne spotkania

- 19 kwietnia - Kontynuacja zajęć z React Native i Expo
- 26 kwietnia - GraphQL
- 12 maja - Hackathon u nas w biurze



Koniec

Tomasz Sapeta (@tsapeta)

Miron Pawlik (@mironiasty)

Roman Marczewski

github.com/SoftwareMansion/expo-workshops-uj-2018



SOFTWARE MANSION