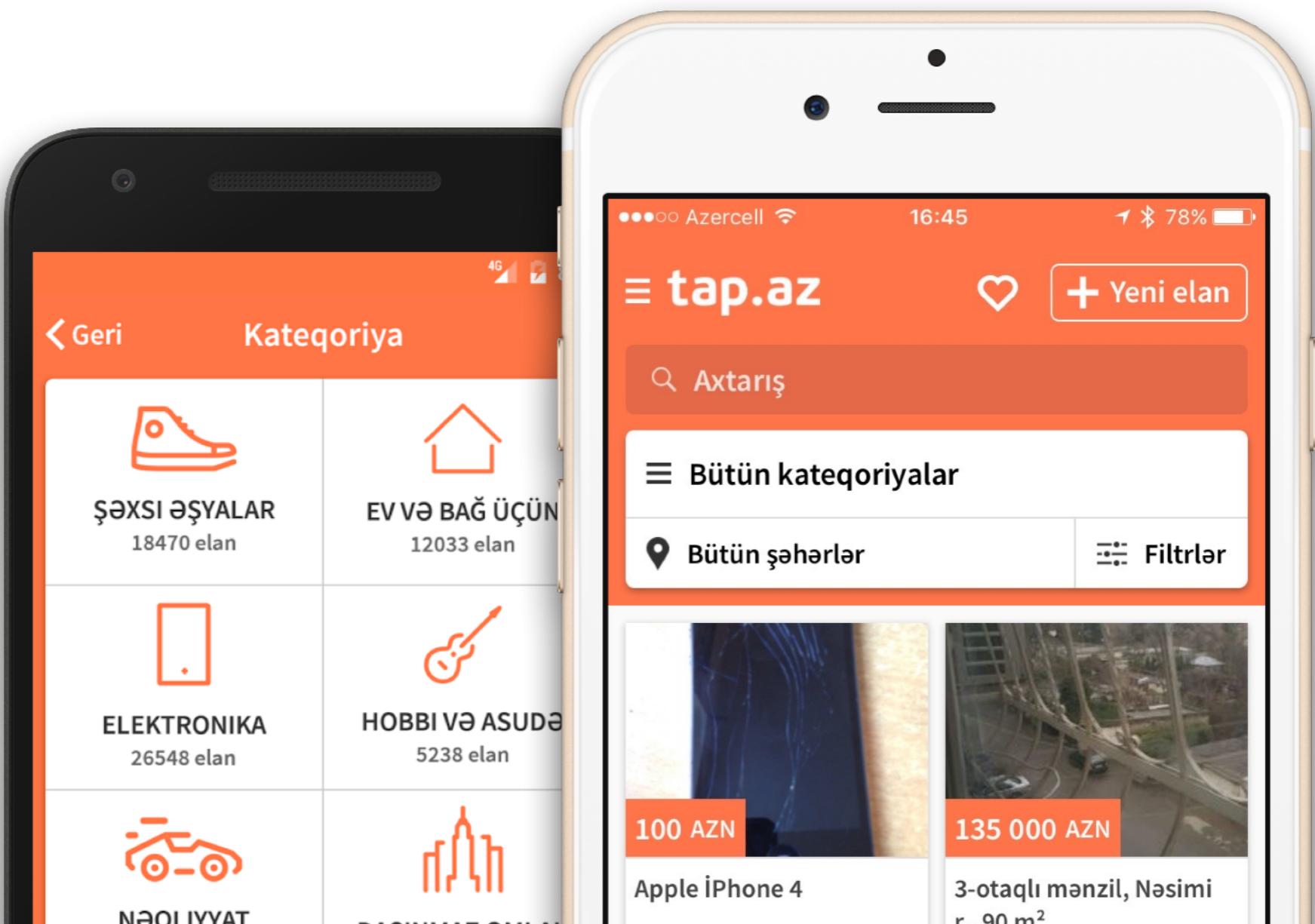


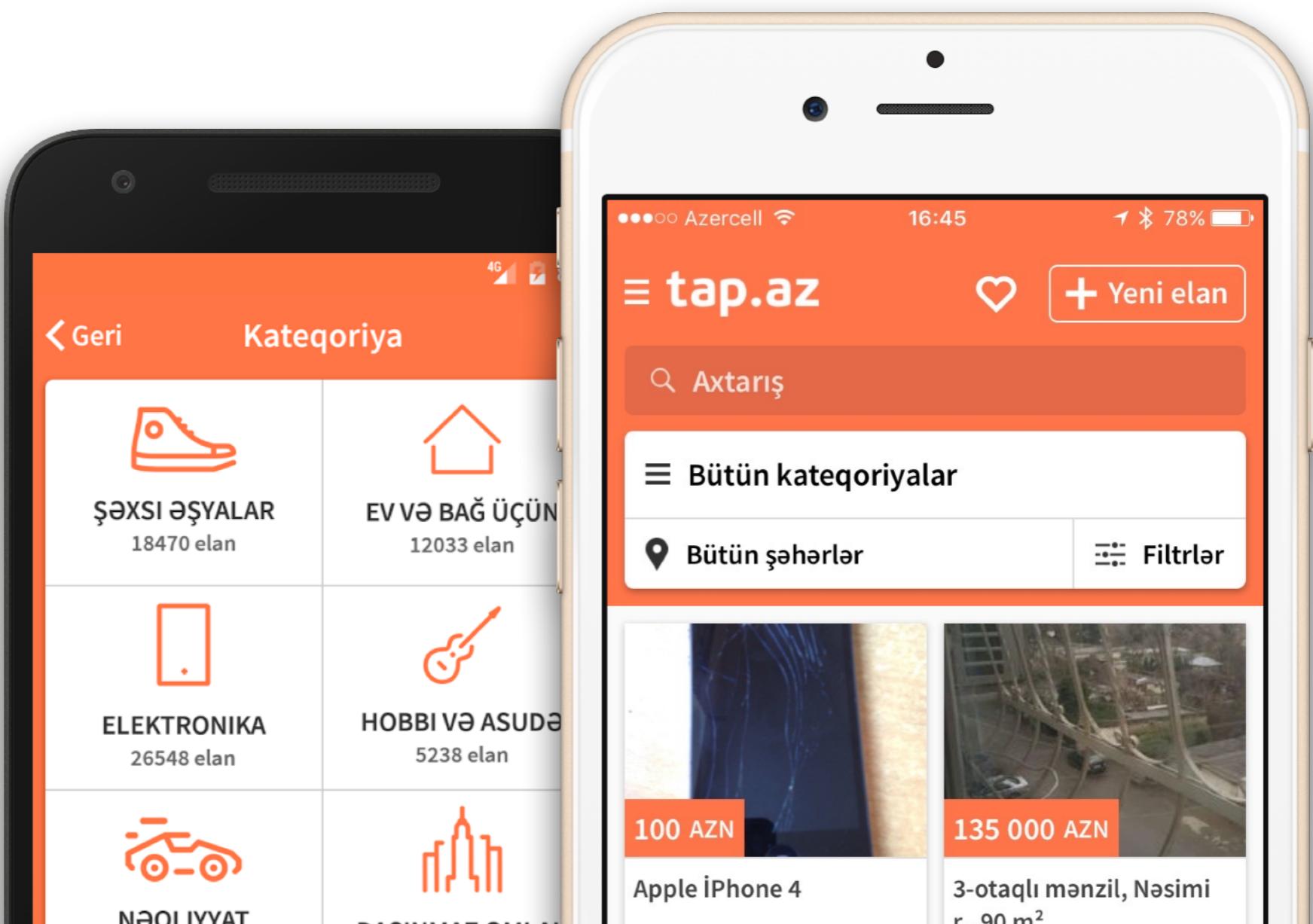
# Using ClojureScript to launch iOS/Android apps to 1M users

by Emin Hasanov  
@hasanove



# Using ClojureScript to launch iOS/Android apps to 1M users

by Emin Hasanov  
@hasanove





Created with mapchart.net ©

# You are here



# You are here



# We are there





# Poland vs Azerbaijan

about 4 times smaller



# Azerbaijan

The Craigslist logo begins with a purple peace sign icon, followed by the word "craigslist" in a lowercase, rounded purple sans-serif font.

tap.az

Əsas səhifə [по-русски](#) [GAP MEDIA – saytda reklam](#)

Bütün Azərbaycan [Bütün elanlar](#)

[+ ELAN YERLƏŞDİR](#)

Məsələn, iPhone 7

TAP [Bütün elanlar](#)

[Elektronika](#) [Daşınmaz əmlak](#) [Şəxsi əşyalar](#) [Nəqliyyat](#)

[Ev və bağ üçün](#) [İş və biznes](#) [Hobbi və asudə](#) [Heyvanlar](#)

1 222 000 istifadəçi gün ərzində

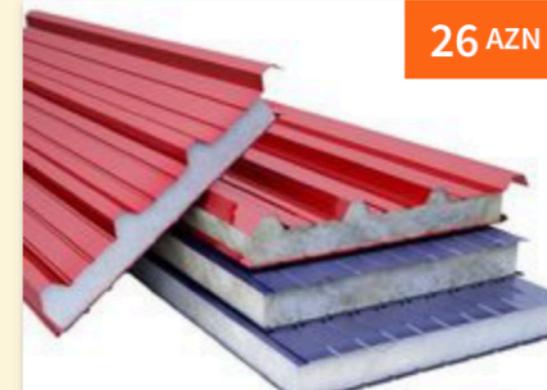
2 515 000 baxış gün ərzində

8 044 yeni elan gün ərzində

VIP ELANLAR [Hamisini göstər](#)

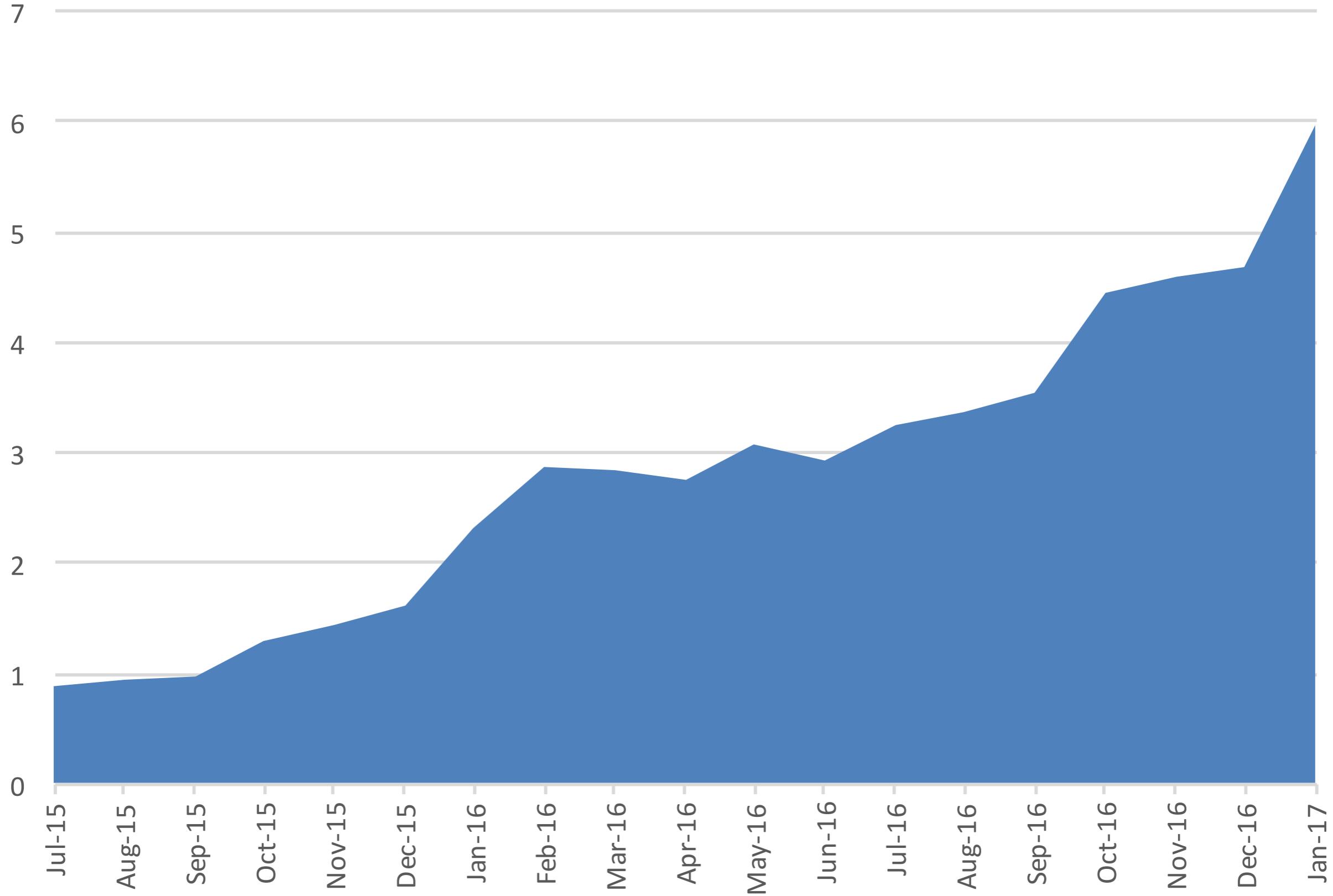
 19 000 AZN

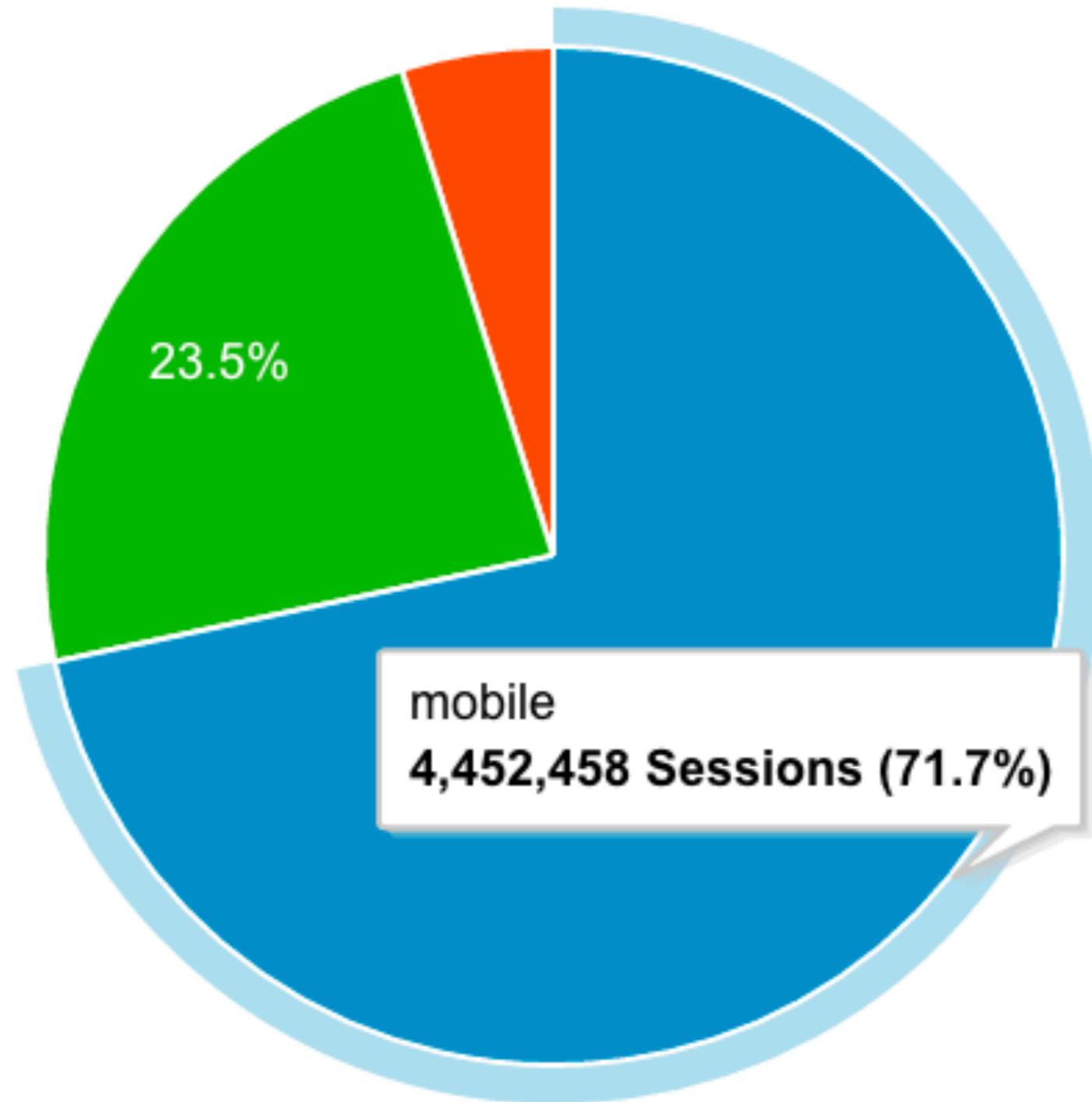
 90 AZN

 26 AZN

 360 AZN

## Monthly sessions (millions)





# Key criteria

Be able to launch and iterate quickly

Maintain same codebase for iOS/android apps

Avoid having to use Objective-C or Java

Have access to platform native features

Get fast, native experience on both platforms

## **50 shades of hybrid**



Website

Native app

# Website wrapper

Quick and easy, same codebase

Worse experience than website, because of user expectations

Does not add much value, except for desktop icon

Yet, not to be underestimated

# Semi-native

Part native, part web views

Easy to start, lots of code sharing with website

Need to know native platforms

Slower than native apps for many cases

App store updates still required for any change to native code

# Almost native

React Native / Xamarin / NativeScript / Others

Develop using same tools, get (almost) native apps

Most of the code shared between ios/android

Most updates do not require app stores

# React Native

Great experience using React on the web

Good support for both iOS and Android

Mature ecosystem and community

Very easy to get started and build apps

Code in JavaScript 😊

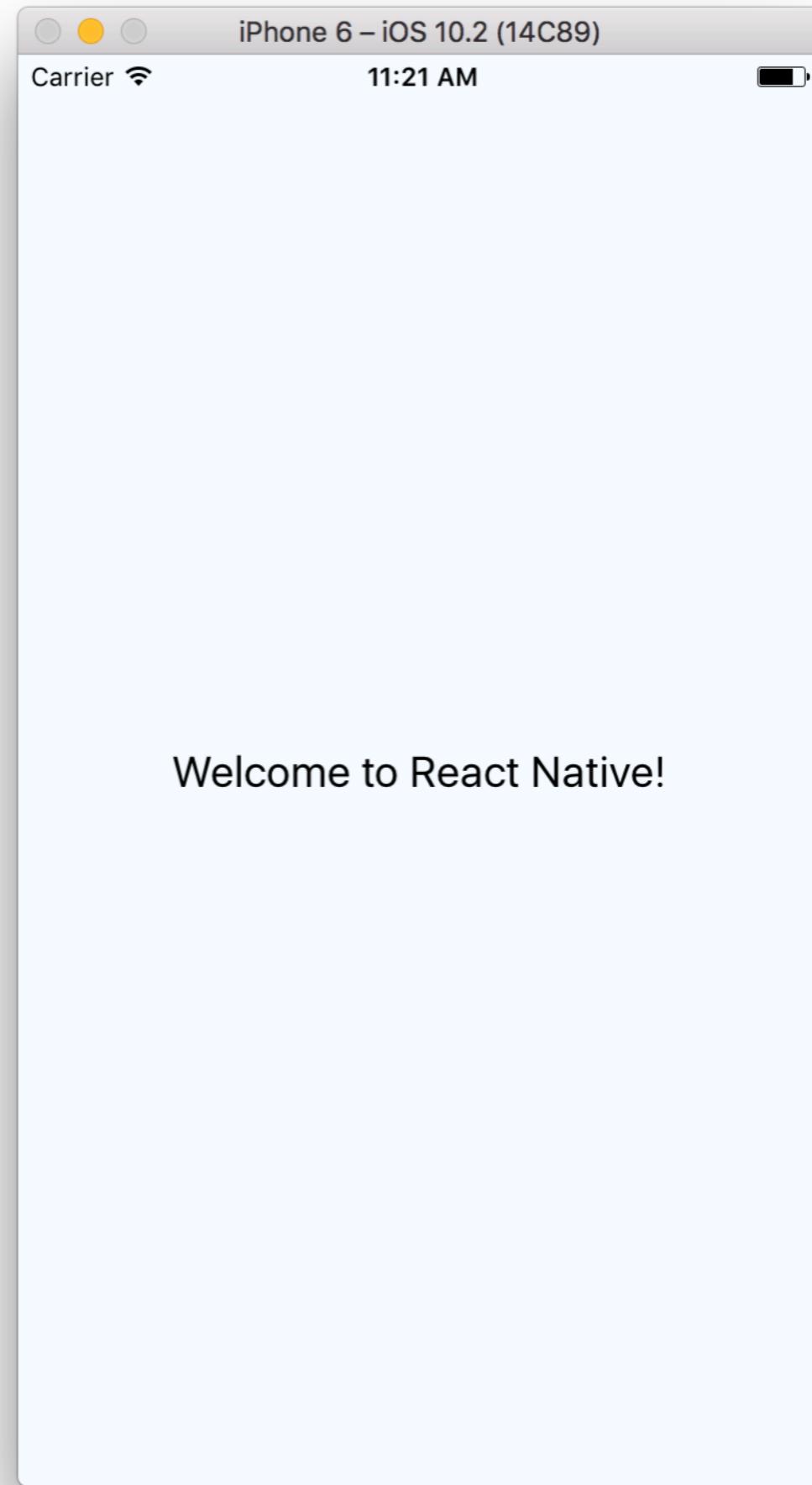
# Getting started

```
$ npm install -g react-native-cli
```

```
$ react-native init FutureApp
```

```
$ cd FutureApp
```

```
$ react-native run-ios
```



```
import React, { Component } from 'react';
import { AppRegistry, StyleSheet, Text, View } from 'react-native';

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
    backgroundColor: '#F5FCFF',
  },
  welcome: {
    fontSize: 20,
    textAlign: 'center',
    margin: 10,
  }
});

export default class Listings extends Component {
  render() {
    return (
      <View style={styles.container}>
        <Text style={styles.welcome}>Welcome to React Native!</Text>
      </View>
    );
  }
}

AppRegistry.registerComponent('Listings', () => Listings);
```

# ClojureScript

Robust, modern, beautiful language

Immutability and concurrency as core concepts

Rich set of data structures

Interactive development with hot code reloading

Lisp!

# ClojureScript for Skeptics

by Derek Slager

<https://www.youtube.com/watch?v=gsffg5xxFQI>

# A taste of ClojureScript

```
{let {x, y, ...vals} = {x: 1, y: 2, z: 3, _: 4};  
  let nums =  
    () =>  
      (function* (xs) {  
        for (let i of xs)  
          {let [arg1, arg2] = xs;  
           yield i ** (arg1 + arg2)}  
           yield* (for (c of [4,5,6]) if (c!==4) c**c);  
      })  
      ([1,2,3]);  
      for (let n of nums()) ::console.log('n:' + n);  
      ::console.log([...nums()]);}
```

# Simple Made Easy

by Rich Hickey

<https://www.infoq.com/presentations/Simple-Made-Easy>

# Example component

```
(defn simple-component []
  [:div
    [:p "I am a component!"]
    [:p.someclass
      "I have " [:strong "bold"]
      [:span {:style {:color "red"}}] " and red "] "text."])
```

## Example

I am a component!

I have **bold** and **red** text.

# Another component

```
(ns example
  (:require [reagent.core :as r]))
(def click-count (r/atom 0))

(defn counting-component []
  [:div
    "The atom " [:code "click-count"] " has value: "
    @click-count ". "
    [:input {:type "button" :value "Click me!"
             :on-click #(swap! click-count inc)}]])
```

Example

The atom click-count has value: 3.

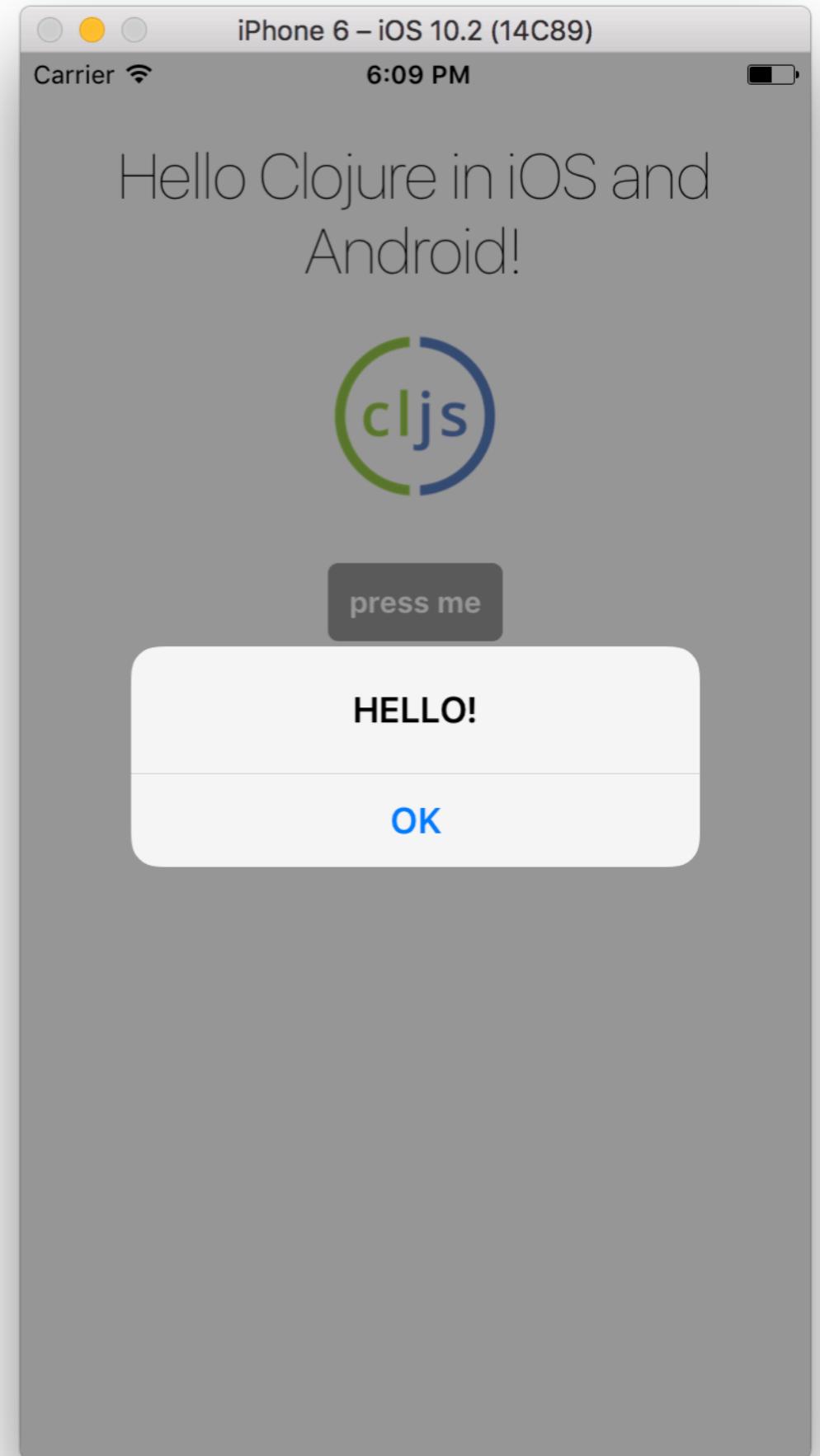
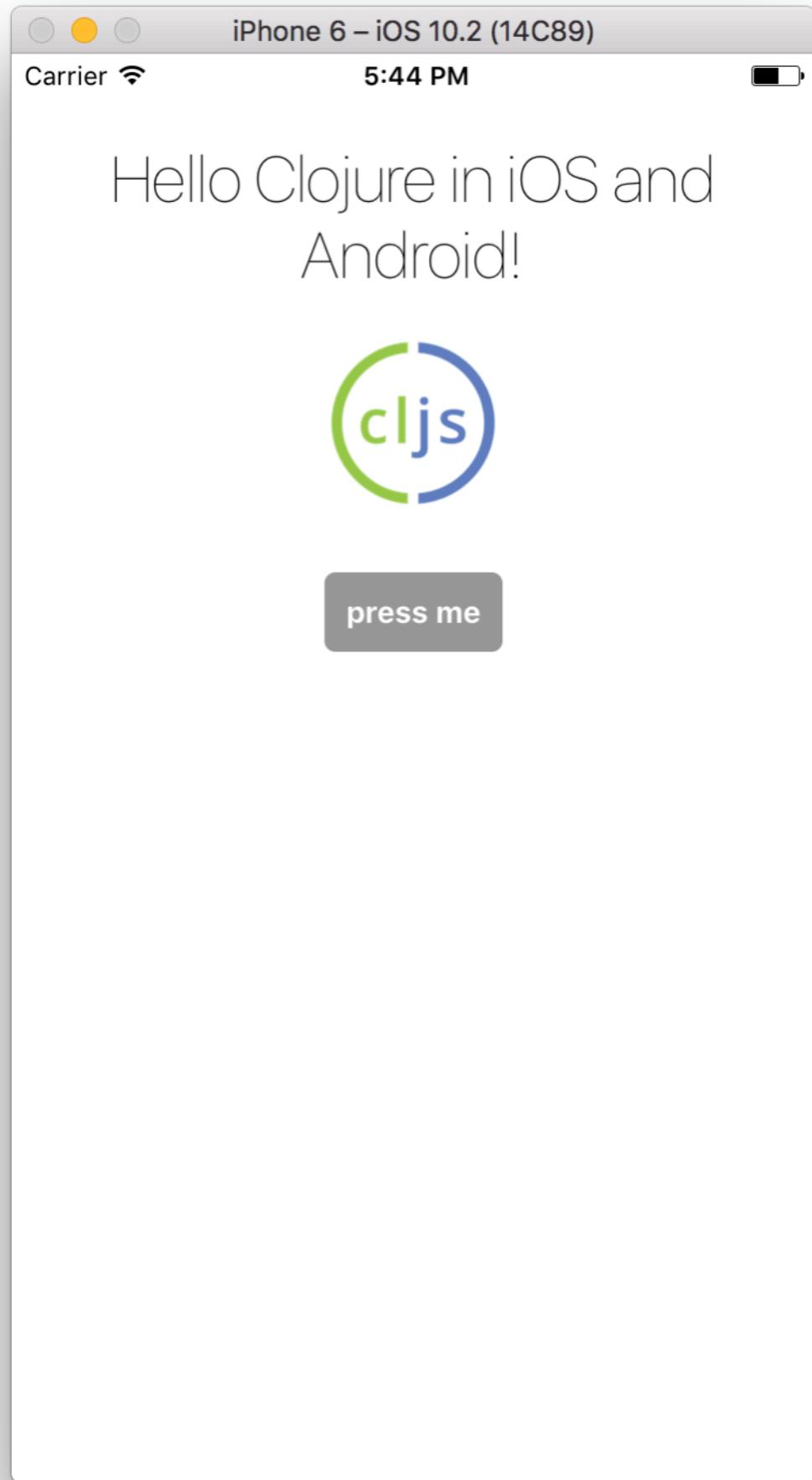
# Getting started

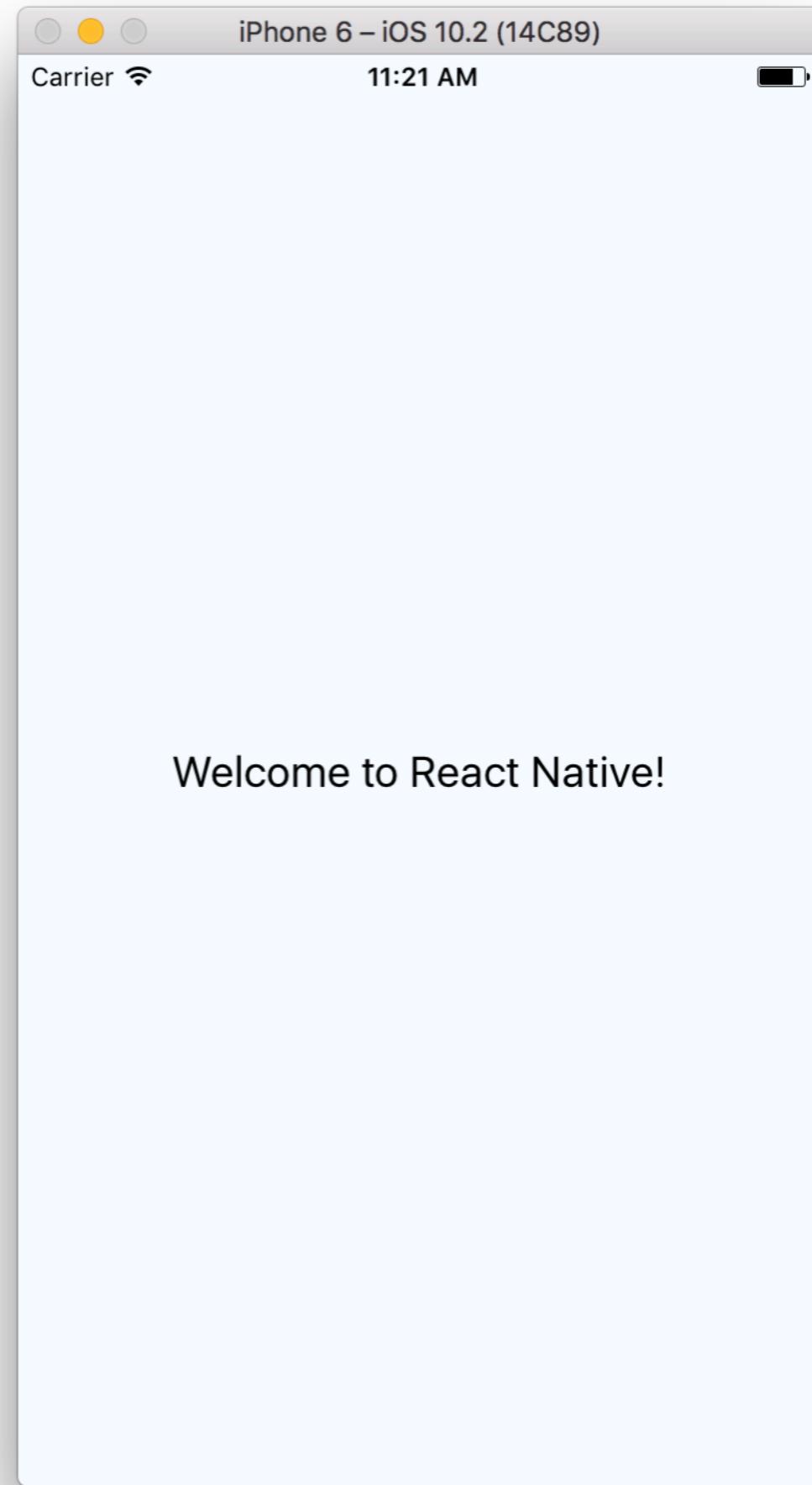
```
$ npm install -g re-natal
```

```
$ re-natal init FutureApp
```

```
$ cd future-app
```

```
$ react-native run-ios
```





```
(ns listings.ios.core
  (:require [reagent.core :as r]))

(def ReactNative (js/require "react-native"))
(def app-registry (.AppRegistry ReactNative))
(def text (r/adapt-react-class (.Text ReactNative)))
(def view (r/adapt-react-class (.View ReactNative)))

(def styles {:container {:flex 1
                         :justifyContent "center"
                         :alignItems "center"
                         :backgroundColor "#F5FCFF"}

             :welcome {:fontSize 20
                       :textAlign "center"
                       :margin 10}})

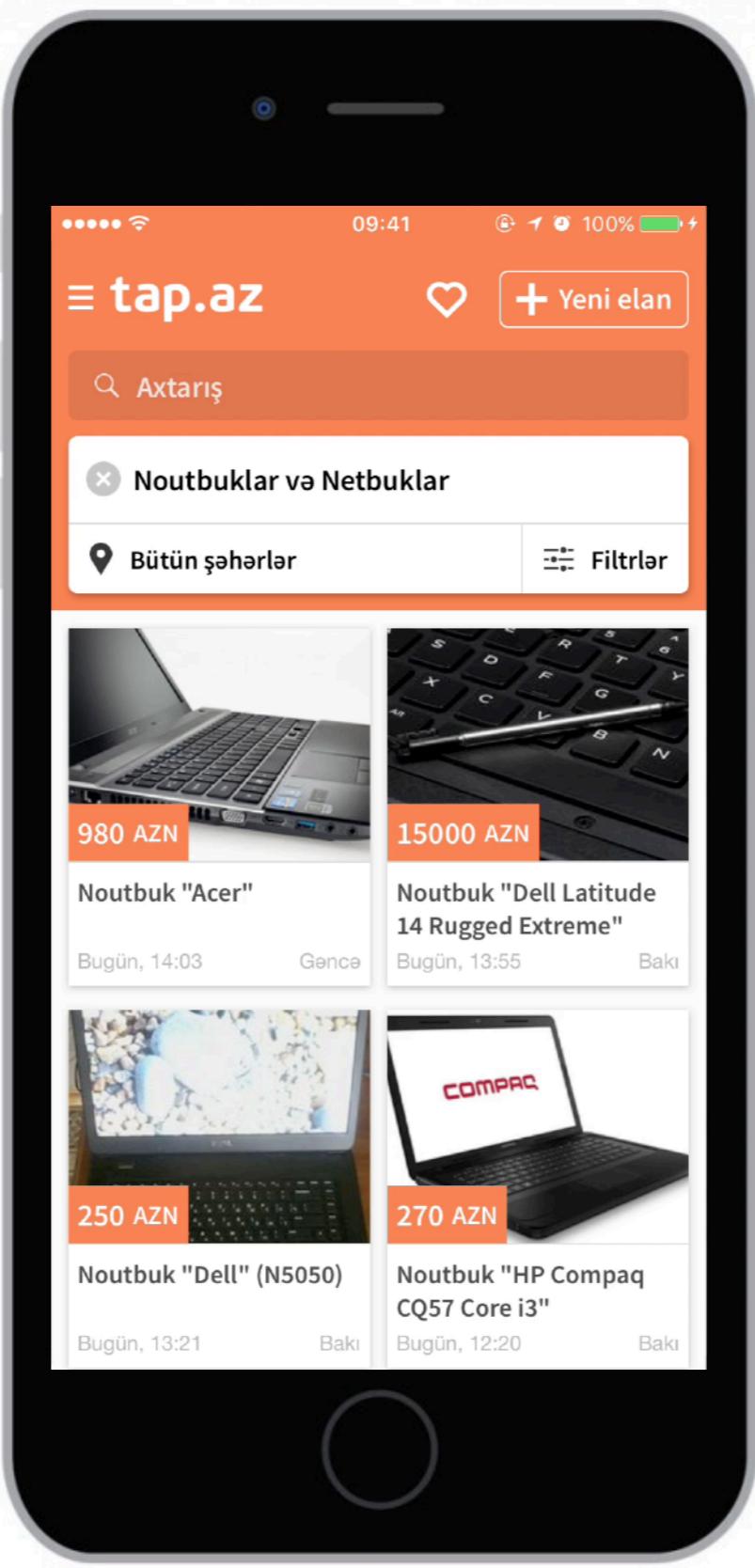
(defn app-root []
  [view {:style (styles :container)}
   [text {:style (styles :welcome)} "Welcome to React Native!"]])

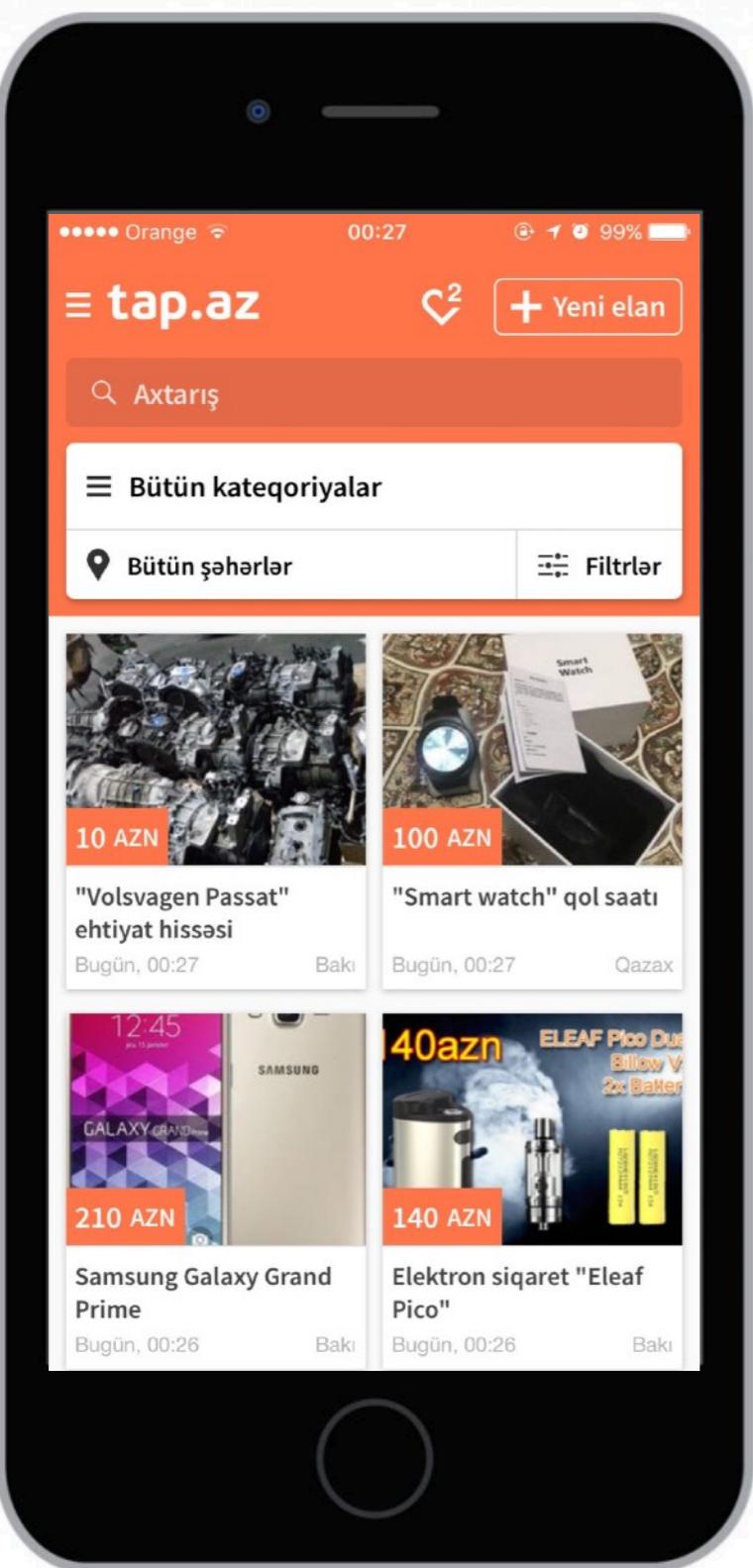
(defn init []
  (.registerComponent app-registry "listings" #(r/reactify-component app-root)))
```

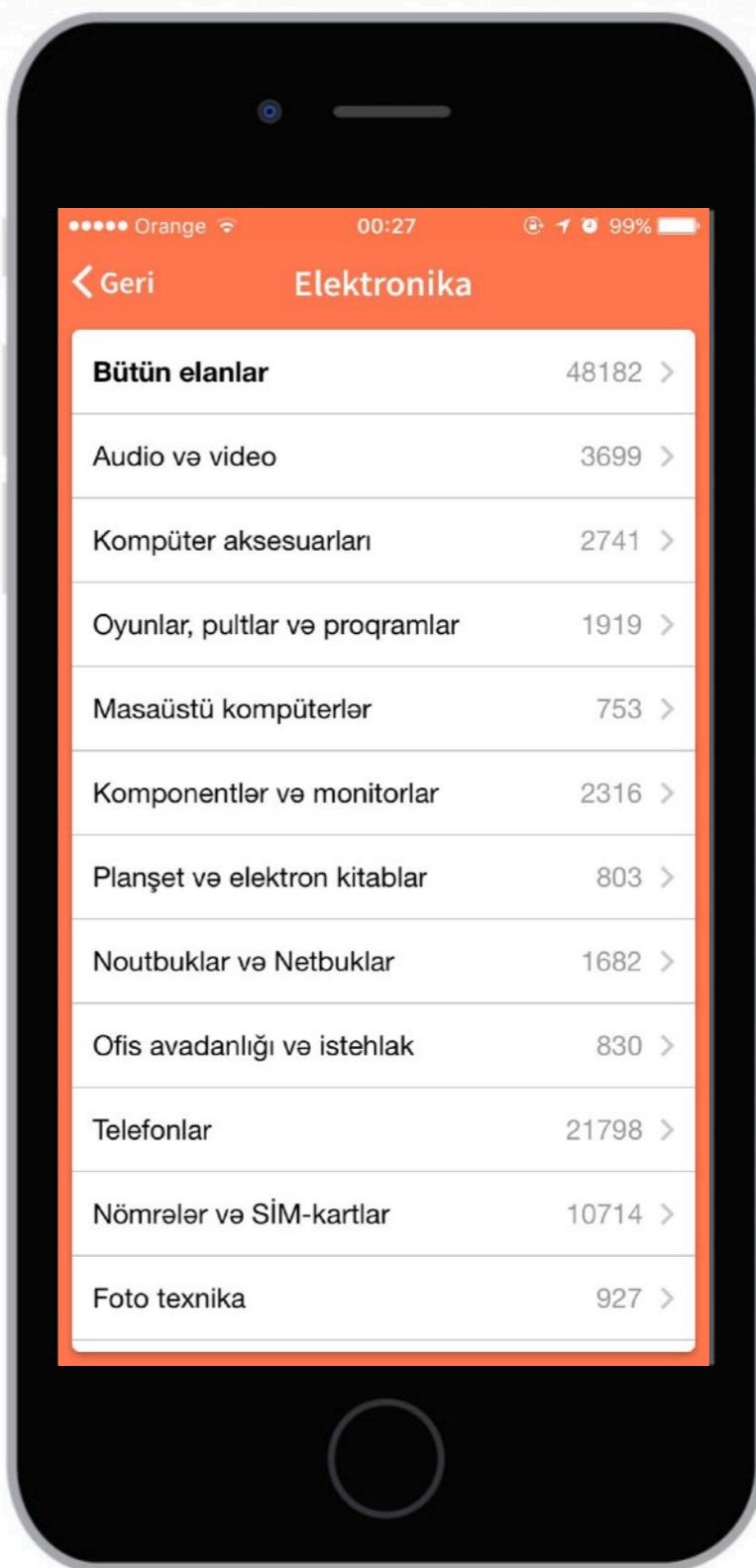
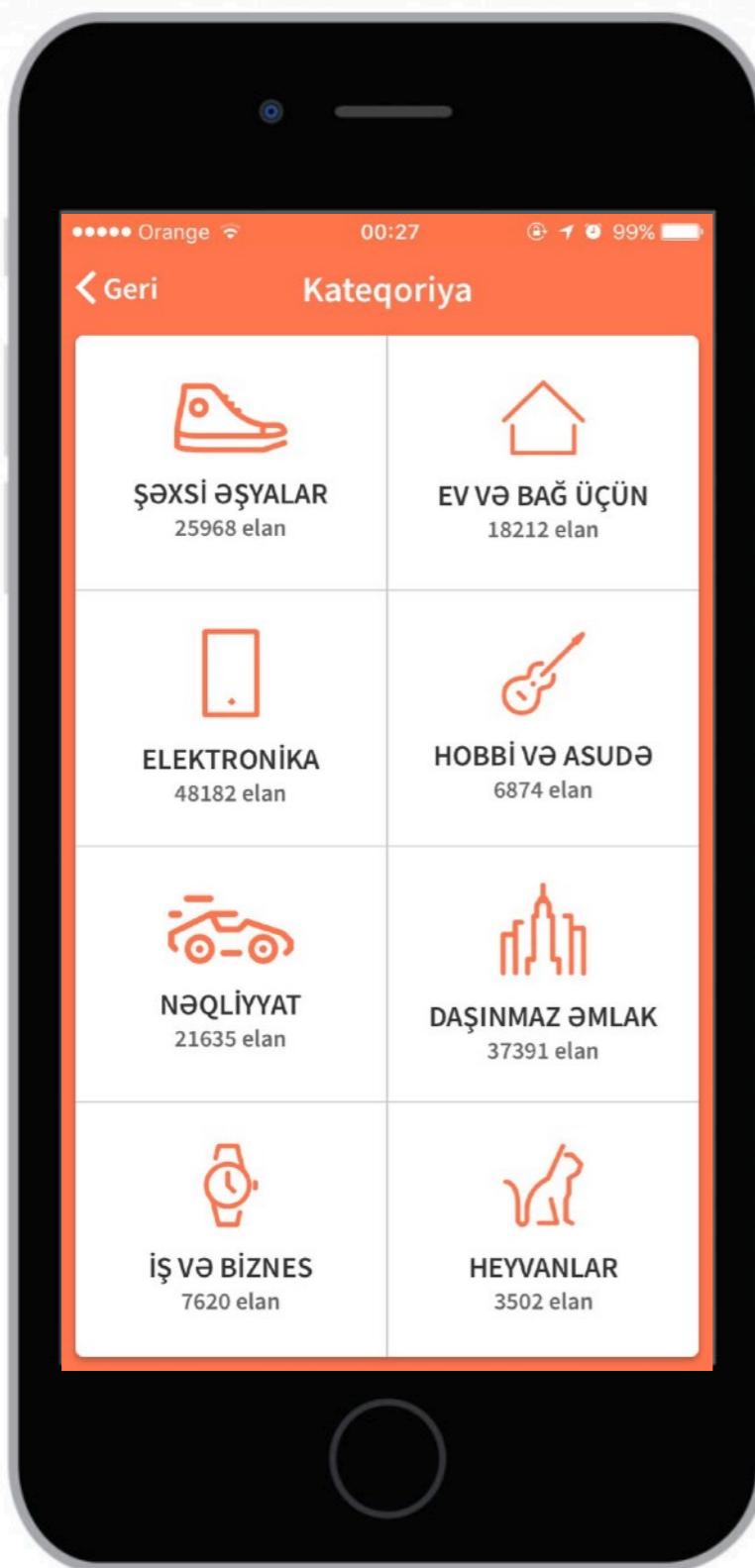
```
export default class Listings extends Component {
  render() {
    return (
      <View style={styles.container}>
        <Text style={styles.welcome}>Welcome to React Native!</Text>
      </View>
    );
  }
}
```

```
(defn app-root []
  [view {:style (styles :container)}
   [text {:style (styles :welcome)} "Welcome to React Native!"]])
```

# Demo







# Transit format

# Transit format

Works on top of JSON (and MessagePack)

Supports any data type, many built-in

Arbitrary keys, not just strings

Language-agnostic

# Transit example

```
{:key => value} # Hash in Ruby
```

```
{:key value} # Map in Clojure
```

# Updates over the air

# CodePush

Register and integrate SDK

Push updates to server

Download in background when app is used

Apply when convenient

Rollback in case of problems

# Update over the air

```
$ code-push deployment list
```

```
$ code-push release-react
```

```
$ code-push deployment history
```

```
$ code-push rollback
```

# Results

# Results

Launched 6 months ago, iOS 8+, Android 4.1+

More than 100,000 installs

15% of the daily sessions are from the app

99.6% crash-free sessions on both platforms

Average rating is above 4.6 (from ~1000 reviews)

**“I didn’t like the app, but I’ll give  
you ★★★★☆ for the effort”**

Thank you!

# Questions?