

AN INTRO TO

REACT NATIVE

WHAT ARE WE HERE FOR?

- ▶ Overview
- ▶ Diving In
 - ▶ Pros and Cons
 - ▶ How Does It Work?
- ▶ Tempt the Demo Gods
- ▶ Wrap-Up



CRAIG
SAGER



REPORTING

CHI 96

MIA 66

4/30/24

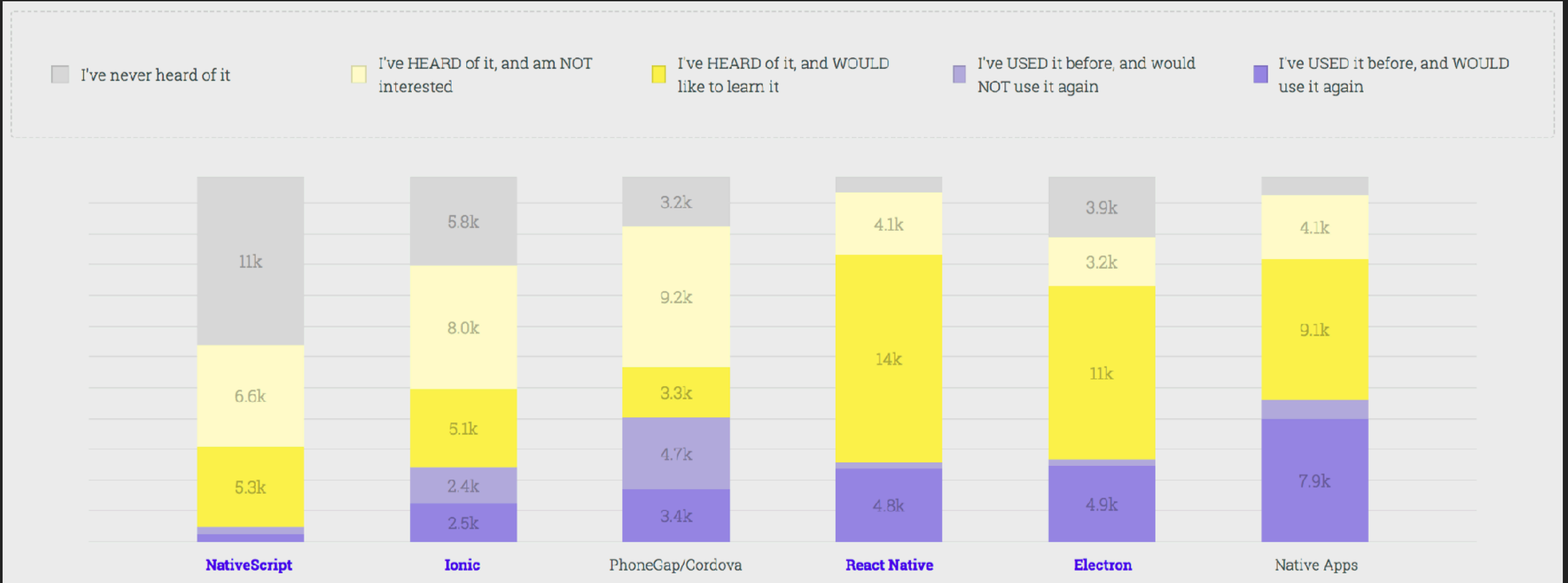
YOU ARE WATCHING THE CHICAGO BULLS VS MIAMI HEAT

NBA ON TV

www.JJ.AM

OVERVIEW

- ▶ React Native lets you build mobile apps using only JavaScript.
- ▶ Not a hybrid/webview app, uses native UI views
- ▶ Share business logic
- ▶ Support for Android, iOS (iPhone + iPad), Android TV, and tvOS (+ Microsoft?!)
- ▶ MIT License



<https://stateofjs.com/2017/mobile/results>

WHO USES REACT NATIVE?

- ▶ Facebook + Instagram
 - ▶ Created React Native, Ads Manager, push notifications view
- ▶ AirBnB
 - ▶ Premier map component
- ▶ Walmart
 - ▶ Mobile app, share 96% of codebase
- ▶ Buy-in from others like Microsoft App Center

**DIVING IN TO REACT
NATIVE**

MARK-UP IS VERY SIMILAR TO REACT ON THE WEB

```
1 import React, { Component } from 'react';
2 import { Text, View } from 'react-native';
3
4 class ReactNativeHome extends Component {
5   render() {
6     return (
7       <View>
8         <SomeReallyCoolNativeComponent />
9         <Text>
10          Hello {this.props.name}
11        </Text>
12      </View>
13    );
14  }
15 }
```

```
1 import React, { Component } from 'react';
2
3 class ReactWebHome extends Component {
4   render() {
5     return (
6       <div>
7         <SomeReallyCoolWebComponent />
8         <span>
9           Hello {this.props.name}
10        </span>
11      </div>
12    );
13  }
14 }
```


STYLING IS VERY CSS-LIKE

```
1  const styles = StyleSheet.create({↵
2  ·  container: {↵
3  ·    ·  marginBottom: 24,↵
4  ·    ·  width: '100%',↵
5  ·  },↵
6  ·  header: {↵
7  ·    ·  alignItems: 'center',↵
8  ·    ·  flexDirection: 'row',↵
9  ·    ·  justifyContent: 'flex-start',↵
10 ·    ·  marginBottom: 8,↵
11 ·  },↵
12 ·  label: {↵
13 ·    ·  color: colors.darkGray,↵
14 ·    ·  fontSize: 20,↵
15 ·    ·  fontWeight: '800',↵
16 ·  },↵
17 });↵
```

```
1  .container {↵
2  ·  margin-bottom: 24px;↵
3  ·  width: 100%;↵
4  }↵
5  ↵
6  .header {↵
7  ·  align-items: 'center',↵
8  ·  display: flex,↵
9  ·  flex-direction: 'row',↵
10 ·  justify-content: 'flex-start',↵
11 ·  margin-bottom: 8;↵
12 }↵
13 ↵
14 .label {↵
15 ·  color: $color-gray-dark;↵
16 ·  font-size: 20,↵
17 ·  font-weight: 800,↵
18 }
```

HOT RELOADING AND CODE PUSH

- ▶ Hot Reloading
 - ▶ Updates to code propagate to the app immediately and automatically
 - ▶ No waiting for app to compile
- ▶ Code Push
 - ▶ Possible to push code updates to store without submission
 - ▶ Depends on your release cadence

WANT NATIVE FUNCTIONALITY? YOU GOT IT!

- ▶ iOS
 - ▶ DatePickerIOS
 - ▶ PushNotificationIOS
- ▶ Android
 - ▶ BackHandler
 - ▶ PermissionsAndroid
- ▶ And more (e.g. camera, clipboard, status bar, web view, etc.)...

**OK, STOP SUGAR-
COATING IT**

ROUTING IS KIND OF A PROBLEM

- ▶ Native
 - ▶ Requires tighter integration
- ▶ JavaScript
 - ▶ Threading issues
- ▶ Developer dependent—*which do you know better?*
- ▶ Many options, switching can be costly

UPDATES CAN FALL BEHIND OS UPDATES

- ▶ At the mercy of development team if you need latest updates
 - ▶ Applies to React Native and Libraries
- ▶ Example 1: iPhone X support was a little slow
- ▶ Example 2: tab height changed in iOS 11

UPDATES ARE PAINFUL

- ▶ Updates to React Native are done through a Git utility
 - ▶ Works...most of the time?
 - ▶ Knowledge of native platforms help
- ▶ No roadmap for React Native
 - ▶ Hard to know what is coming down the pipe
 - ▶ Contributions are welcome to help shape the future

LEARN ONCE, WRITE ANYWHERE

- ▶ NOT Write once, run anywhere
- ▶ UI abstraction can be lossy
- ▶ Not 100% implementation for either platform
- ▶ Many stories of planning to do both platforms and never getting around to one of them

HOW DOES IT WORK?

- ▶ Two threads
 - ▶ Main - User interface and gestures, e.g. UIKit
 - ▶ JS - JS code and business logic
- ▶ Don't communicate, use a bridge
- ▶ 60 FPS is hard sometimes
 - ▶ If JS is blocked, some UI is unresponsive (e.g. navigation, or TouchOpacity)


DEMO





WRAP-UP

COMPETITION

- ▶ Flutter
 - ▶ Google, Dart, very new, handle UI differently
- ▶ Expo
 - ▶ Toolchain built on React Native, great for simple apps
- ▶ Cordova/Hybrid Apps
 - ▶ 

SO...

- ▶ Will not fix problems stemming from process or staffing
- ▶ Think about what you're making
- ▶ Test on real devices!
- ▶ Can always fall back to native components and helpers
- ▶ Things change quickly! 🎉😭
- ▶ Good for JS devs with interest in native or native devs with interest in JS

WE DONE YET?



CRAIG
SAGER



REPORTING

CHI 96

MIA 66

4/30/24

YOU ARE WATCHING THE CHICAGO BULLS VS MIAMI HEAT

NBA ON TV

www.JJAM

QUESTIONS?



LINKS

- ▶ <http://artsy.github.io/blog/2018/03/17/two-years-of-react-native/>
- ▶ https://www.reddit.com/r/reactnative/comments/8a8vgj/why_did_you_choose_react_native_what_dodont_you/
- ▶ <https://medium.com/@talkol/performance-limitations-of-react-native-and-how-to-overcome-them-947630d7f440>
- ▶ <https://brainhub.eu/blog/famous-apps-built-with-react-native/>
- ▶ <https://levelup.gitconnected.com/wait-what-happens-when-my-react-native-application-starts-an-in-depth-look-inside-react-native-5f306ef3250f>
- ▶ <https://blog.usejournal.com/thoughts-about-react-native-after-a-few-months-working-with-it-4b3e255c3120>
- ▶ <https://medium.com/airbnb-engineering/react-native-at-airbnb-f95aa460be1c>