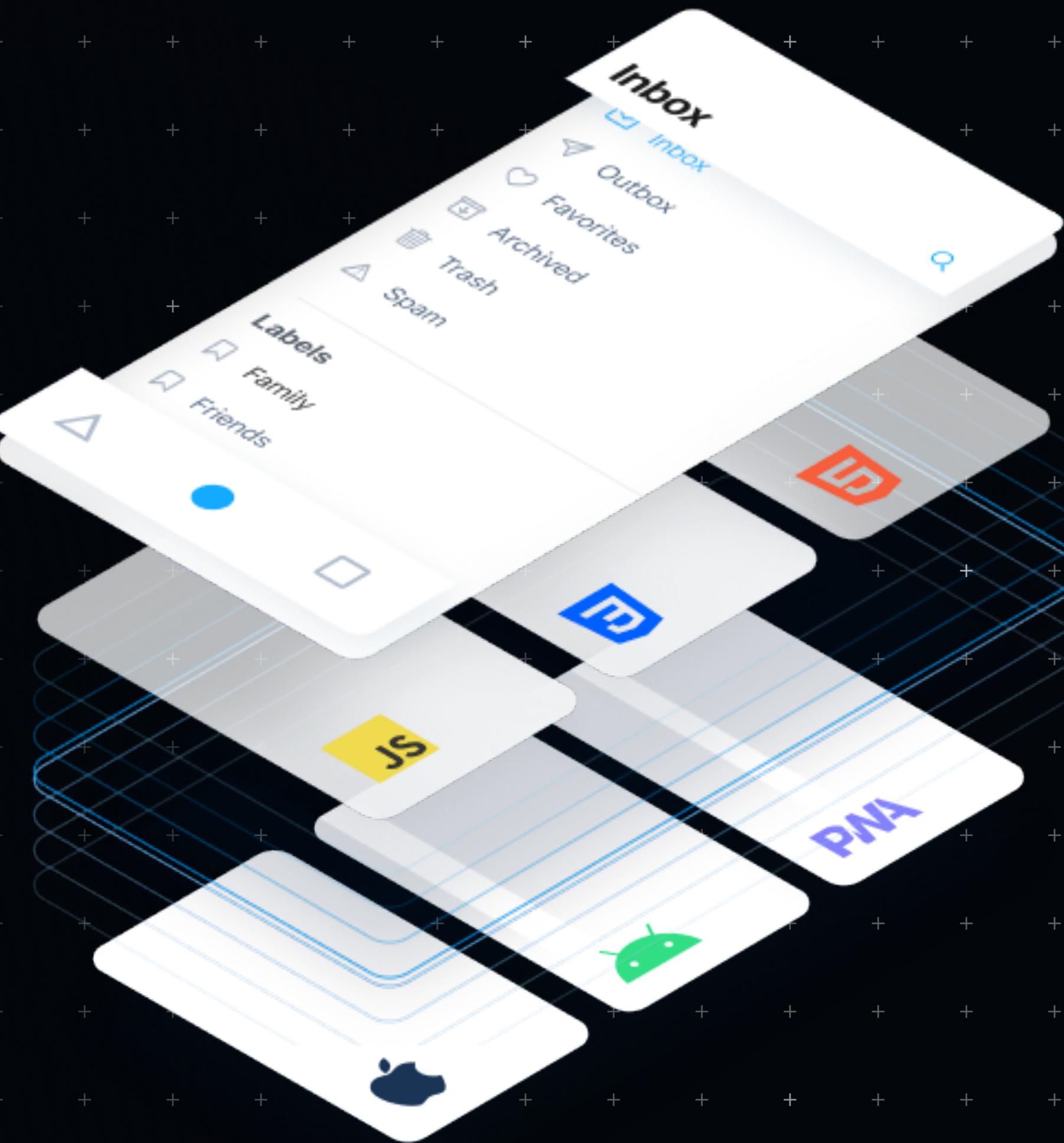




One App Multiple Platforms

How Cross Platform Actually Works

Mike Hartington | @mhartington



Cross Platform Development



Sorcerer Rogier



Wild Strikes



25

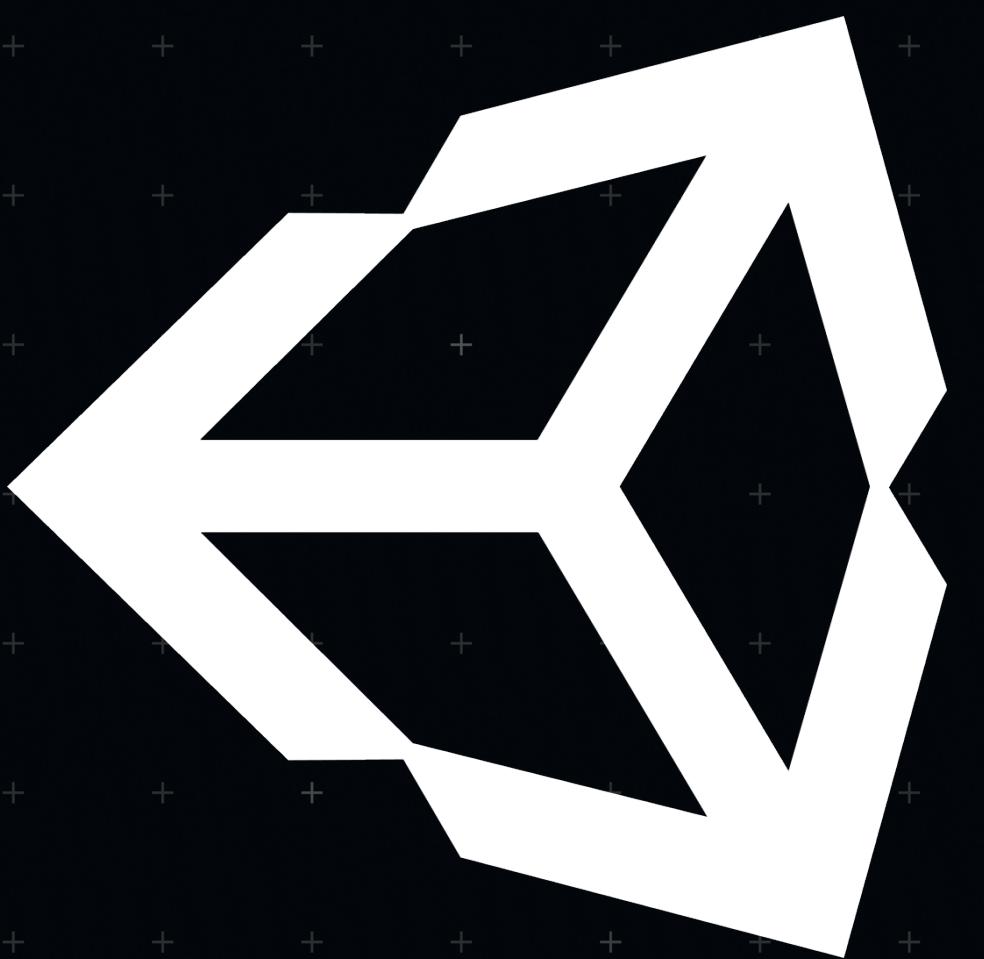
Bone Dart

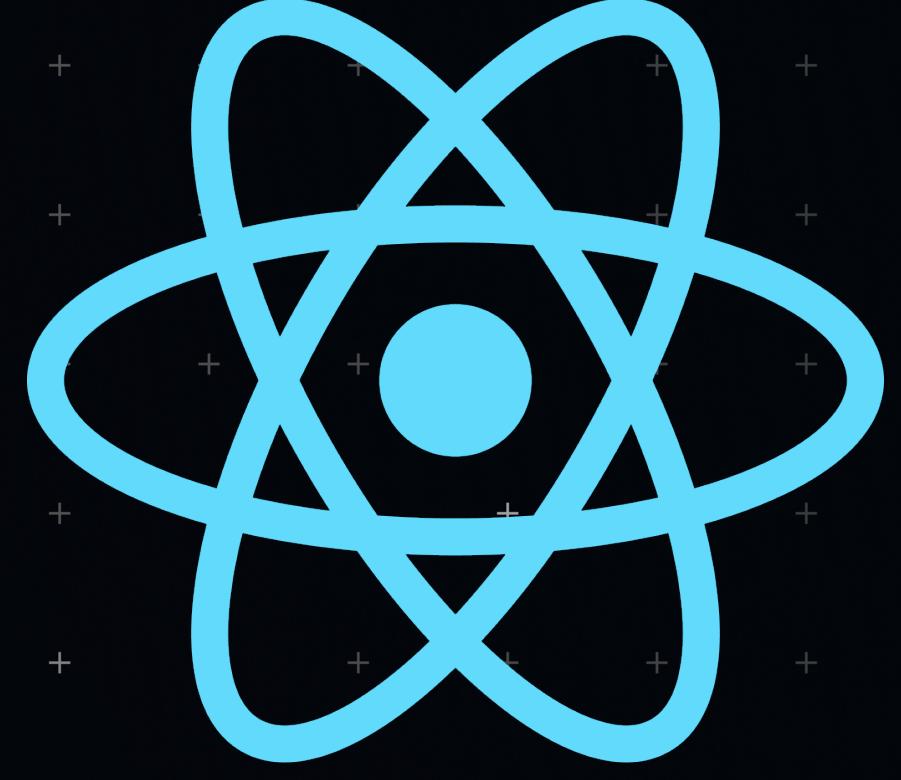
Margit, the Fell Omen

Y : Touch bloodstain



0





**How do these
options compare**

**What do the concern
themselves with**

Project Setup

Programming Language

Navigation/Routing

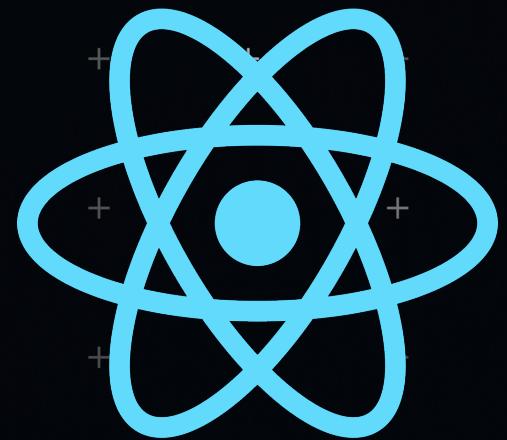
Third-party Libraries

Native API access

Web Deployment

Tooling & Debugging

UI/Design Features



React Native

bit.ly/3y113GO



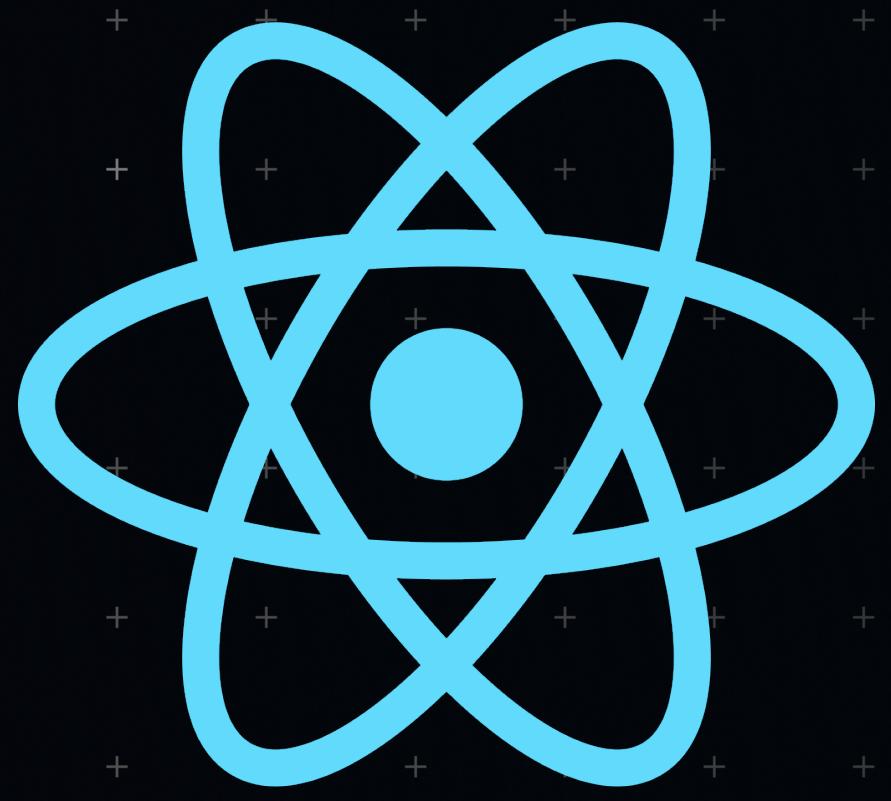
ionic

bit.ly/3fuKWef

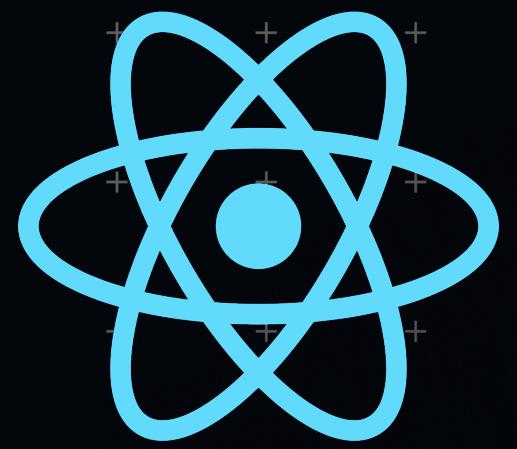


Flutter

bit.ly/3RIQTaD



React Native



React Native

Declarative API for native controls

Abstracts iOS and Android APIs

JSX for authoring

"Learn once, write anywhere."

To Expo or Not Expo

\$ npx create-expo-app my-app

\$ npx react-native init my-app

To Expo or Not Expo

\$ npx create-expo-app my-app

\$ npx react-native init my-app

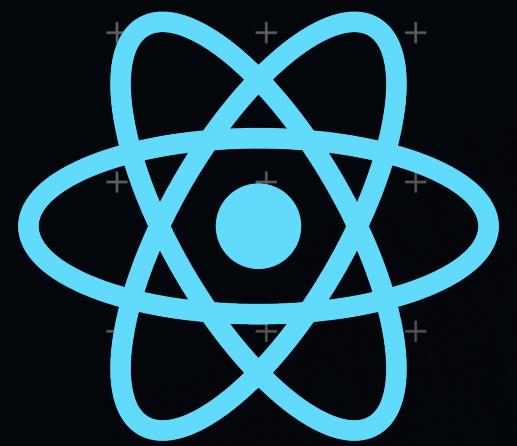
Probably just use expo?



The screenshot shows a mobile application development interface. On the left, the code editor displays `App.js` with the following content:

```
1 import { StatusBar } from 'expo-status-bar'
2 import { StyleSheet, Text, View }
3
4 export default function App() {
5   return (
6     <View style={styles.container}>
7       <Text>Open up App.js to start working on your app!
8       <StatusBar style="auto">
9     </View>
10  );
11}
12
13 const styles = StyleSheet.create({
14   container: {
15     flex: 1,
16     backgroundColor: '#fff',
17     alignItems: 'center',
18     justifyContent: 'center',
19   },
20 });
21
```

The right side shows a preview of the application running on an iPhone 14 Pro with iOS 16.0. The screen displays the text "Open up App.js to start working on your app!" centered in a white view. The top status bar shows the time as 5:42 and various connectivity icons.



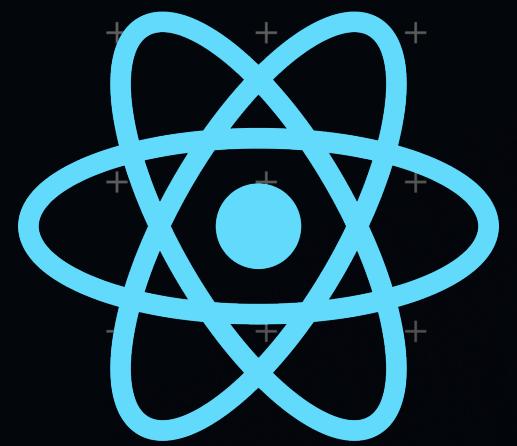
React Native - Language

JS Out of the box

TypeScript?

```
touch tsconfig.json && npx expo start
```

Rename files to `.



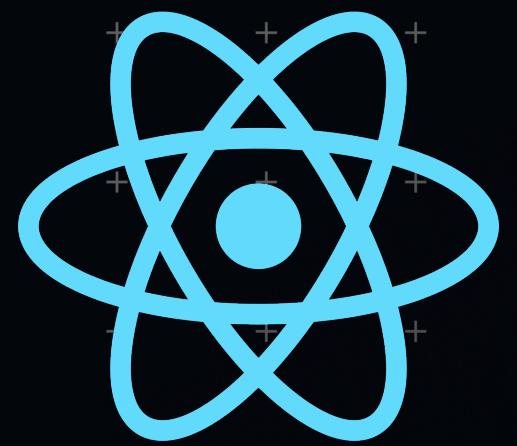
React Native - Routing

Did you pick expo?

@react-navigation/native, plus others

Additional support packages needed

Else react-native-navigation



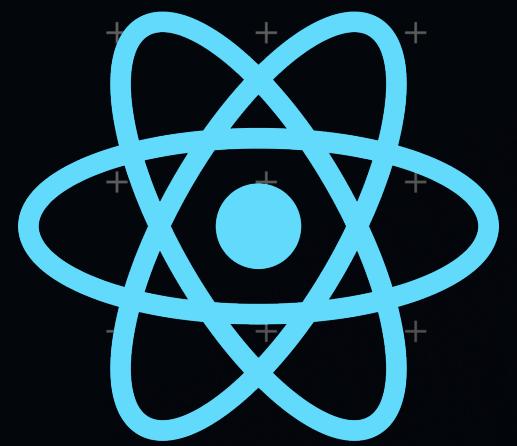
React Native - Third-party Libraries

Did you pick expo?

Search for specific React-Native libs

Some are platform specific

Existing tools are a no-go



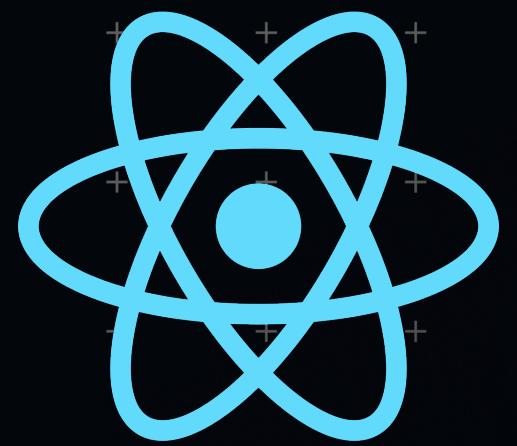
React Native - Native API Access

Did you pick expo?

Core features provide base

Not available? Build Native Module

Congrats, go write swift/obj-c, java



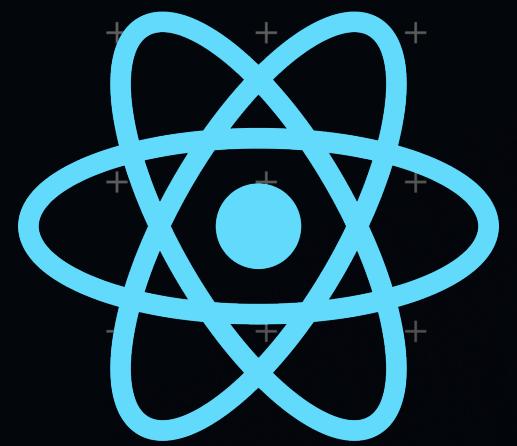
React Native - Web Deployment

Did you pick expo?

`react-native-web` or `react-native-dom`

What libraries did you pick?

Generated DOM is...something



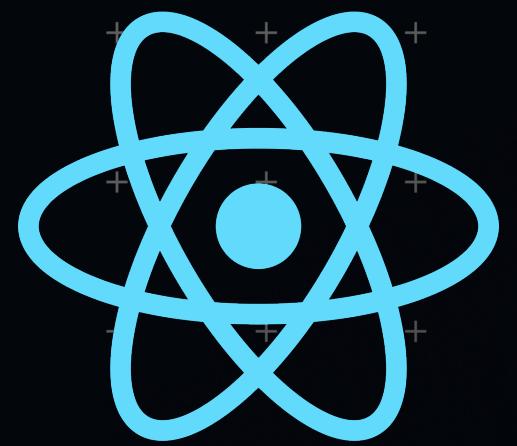
React Native - Tooling & Debugging

Built-in devtools from browsers

Proxies your JS via a web worker

React DevTools work out of the box

Built-in inspector for on device work



React Native - UI/Design Features

Inline styles can get you pretty far

Core StyleSheet provided by React Native

@rneui/themed provides a theming function

Some helper functions to mimic web styles





UI Library and native API

Bindings for Angular, React, Vue

Web Components at the core

iOS, Android, and Web

```
$ npm install -g @ionic/cli
```

```
$ npm install -g @ionic/cli
```

```
$ ionic start my-app
```

```
$ npm install -g @ionic/cli
```

```
$ ionic start my-app
```

Pick a framework! 😊

› Angular | <https://angular.io>

React | <https://reactjs.org>

Vue | <https://vuejs.org>



Default

iPhone 14 Pro
iOS 16.0

8025ms

[ng] ✓ Compiled successfully.

[INFO] Development server running!

Local: http://localhost:8100

Use Ctrl+C to quit this process

> capacitor run ios --no-sync --target D764ABAF-6670-4546-A10F-82DAAB275F

2D

[capacitor] ✓ Running xcodebuild in 19.59s

[capacitor] ✓ Deploying App.app to D764ABAF-6670-4546-A10F-82DAAB275F2D i

n 3.01s

[INFO] App deployed to device!

Development server will continue running until manually stopped.

Use Ctrl+C to quit this process

1. home.page.html

```
1 <ion-header [translucent]="true"
2   <ion-toolbar>
3     <ion-title>
4       Blank
5     </ion-title>
6   </ion-toolbar>
7 </ion-header>
8
9 <ion-content [fullscreen]="true"
10  <ion-header collapse="condense">
11    <ion-toolbar>
12      <ion-title size="large">Blank</ion-title>
13    </ion-toolbar>
14  </ion-header>
15
16  <div id="container">
17    <strong>Ready to create an app?</strong>
18    <p>Start with Ionic <a href="#">Create</p>

```

12:53

Blank

Ready to create an app?
Start with Ionic [UI Components](#)

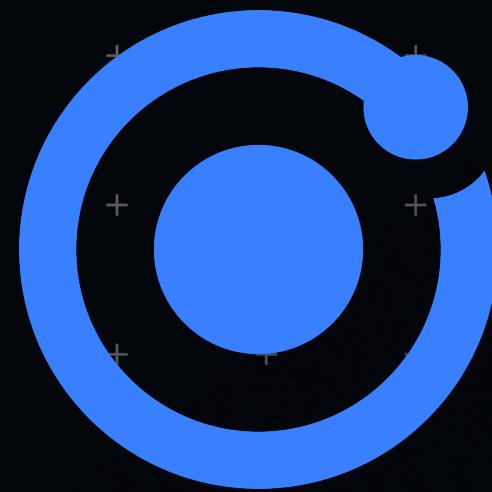
main home.page.html



ionic - Language

TypeScript as the default

Vue & React also use TS



ionic - Routing

Provided by the framework

Stack-based navigation as addition

Support for native style multiple roots

Animations/Transitions provided



ionic - Third-party Libraries

Have you heard of npm?

Pure JS/TS libs or framework specific

Not all libraries are right for mobile

Opportunities to pick wrong



ionic - Native API Access

Provided by Capacitor

Native runtime + Native API

Extensive core feature plus community

Not available? Build a plugin

Congrats, go write swift/obj-c, java



ionic - Web Deployment

Out of the box

All HTML, CSS, JS at the end of the day

Some native features will gracefully fail

Media Queries are your friend



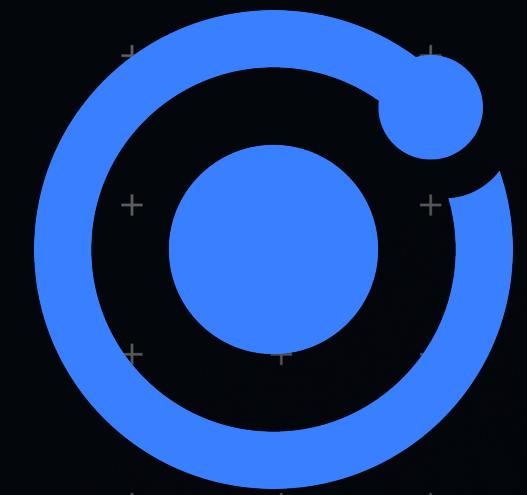
ionic - Tooling & Debugging

Browser DevTools are your friends

Framework specific DevTools also work

Remote debugging is super useful

Live-reload while on deploy to device



ionic - UI/Design Features

CSS/SCSS or what ever you'd like

Core styles are driven by CSS Variables

Theming can scoped to components or global

Modern CSS is pretty flippin awesome



Flutter



Flutter

Open Source framework for X-plat

Written with Dart

Uses custom renderers for UI

iOS, Android, and Web

```
# Install flutter  
$ brew install flutter  
$ flutter create my_app
```

```
"` Github > one-app-multiplatform > my_app
$ flutter run
Launching lib/main.dart on iPhone 14 Pro in debug mode...
Running Xcode build...
└ Compiling, linking and signing... 3.9s
Xcode build done. 21.0s
Syncing files to device iPhone 14 Pro... 107ms

Flutter run key commands.
r Hot reload. 🔥🔥🔥
R Hot restart.
h List all available interactive commands.
d Detach (terminate "flutter run" but leave application running).
c Clear the screen
q Quit (terminate the application on the device).

👉 Running with sound null safety 💪

An Observatory debugger and profiler on iPhone 14 Pro is available at:
http://127.0.0.1:56428/M6tGzxRmHiQ=/
The Flutter DevTools debugger and profiler on iPhone 14 Pro is available at:
http://127.0.0.1:9101?uri=http://127.0.0.1:56428/M6tGzxRmHiQ=/

Default
1. • main.dart
1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(const MyApp());
5 }
6
7 class MyApp extends StatelessWidget {
8   const MyApp({super.key});
9
10 // This widget is the root of your application.
11 @override
12 Widget build(BuildContext context) {
13   return MaterialApp(
14     title: 'Flutter Demo',
15     theme: ThemeData(
16       // This is the theme of your application.
17       // Try running your application using "flutter run -d
18       // application has a blue
19       // changing the primarySwatch to Colors.green instead of
20       // "hot reload" (press "r"
21       // or simply save your changes to "hot reload"
22       // Notice that the counter does not
23       // is not restarted.
24       primarySwatch: Colors.blue,
25     ),
26   ),
27   home: const MyHomePage(title:
28 );
main.dart
2:33
iPhone 14 Pro
iOS 16.0
Flutter App
DEBUG
Flutter Demo Home Page
You have pushed the button this many times:
0
+`
```



Flutter - Language

Dart....it is a language

Actually not that bad

Familiar enough for JS devs

Tooling and editor support is great



Flutter - Routing

Routing follows more native approaches

Have a Navigator widget

Push/Pop base navigation



Flutter - Third-party Libraries

Packages/Plugins for extending core

Published on pub.dev

Sqlite, HTTP, Firebase, etc

A lot of "first" third-party



Flutter - Native API Access

Part of the third-party library

Not part of core

Not available? Build a package

Congrats, go write swift/obj-c, java



Flutter - Web Deployment

Web enabled via flags

Rendering vis WASM

What libraries did you pick?

Generated DOM is...Canvas!



Flutter - Tooling & Debugging

Editor integration is really good

Quality error messages

DevTools are really familiar to chrome

Widget Tree, Memory, Networking, etc



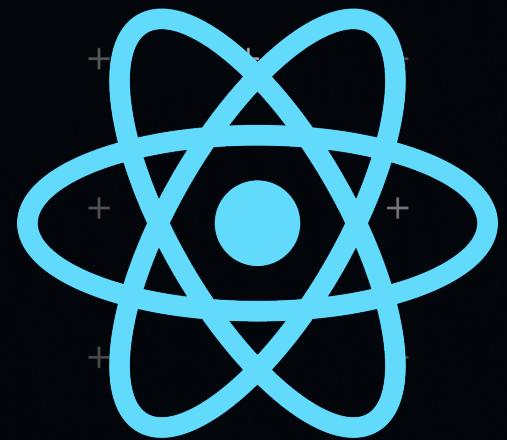
Flutter - UI/Design Features

Core UI provided out of the box

Layout is component base

Theming capabilities via top level config

Depends on Flutter runtime



React Native

bit.ly/3y113GO



ionic

bit.ly/3fuKWef



Flutter

bit.ly/3RIQTaD

Whats the point then?

Most cross-platform tools are similar

Techniques and approaches have come a long way

Ideas have made their way across projects

And performance is not an issue

Not zero sum

Learn from each other

Everyone wins

Questions?