


WEB  DAY

2026

# Rethinking React: Bridging Web to Mobile with React Native

**Violina Popova**  
Software Developer  
ClipMyHorse.TV





**Do you like horses?**

**Do you like horses?**

**But I don't have any  
experience with  
React Native**

**Do you like horses?**

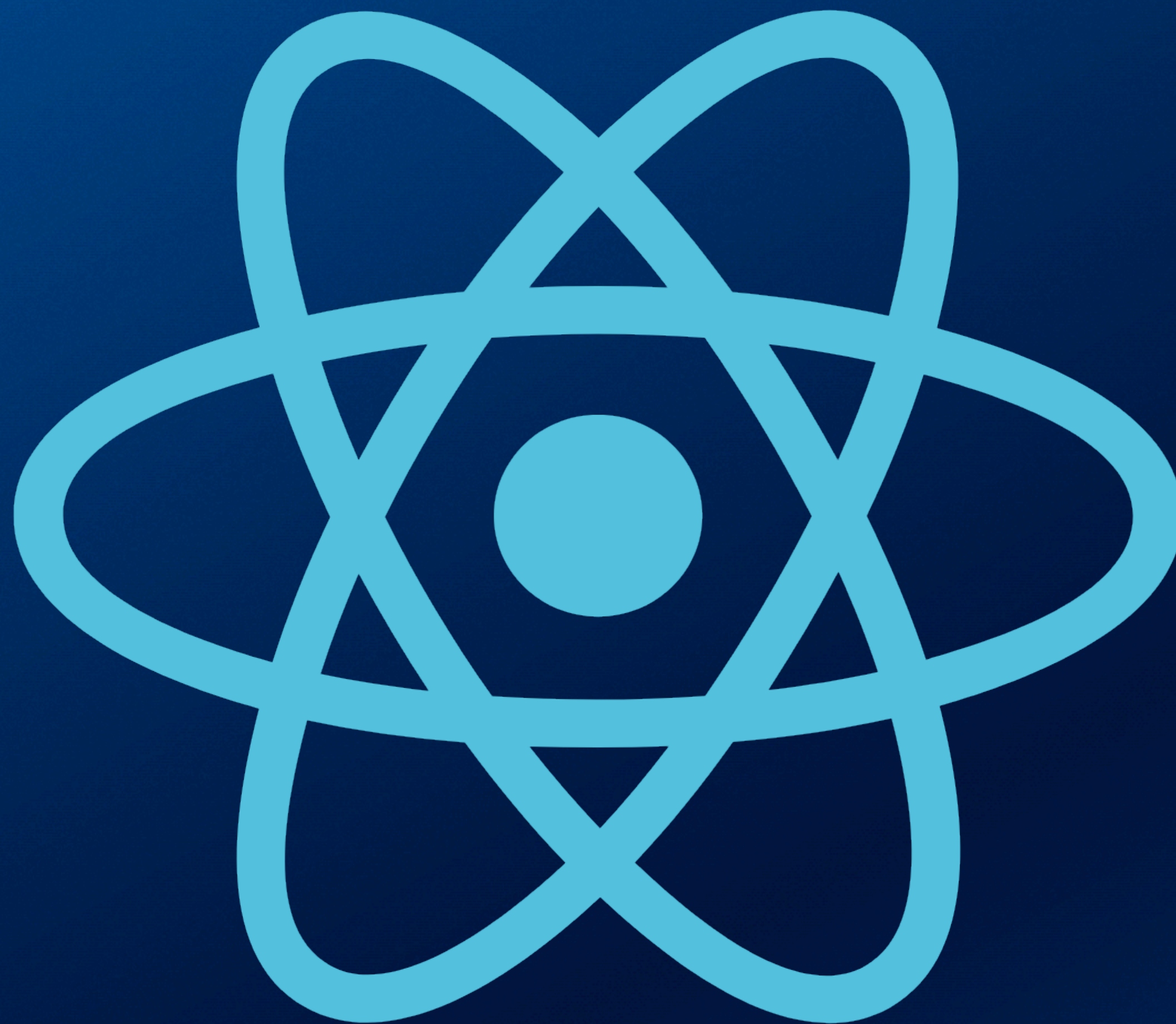
**But I don't have any  
experience with  
React Native**

**You've got React  
experience, that's  
enough**

# My Journey



# My Journey



# A bit about me:



## Violina Popova

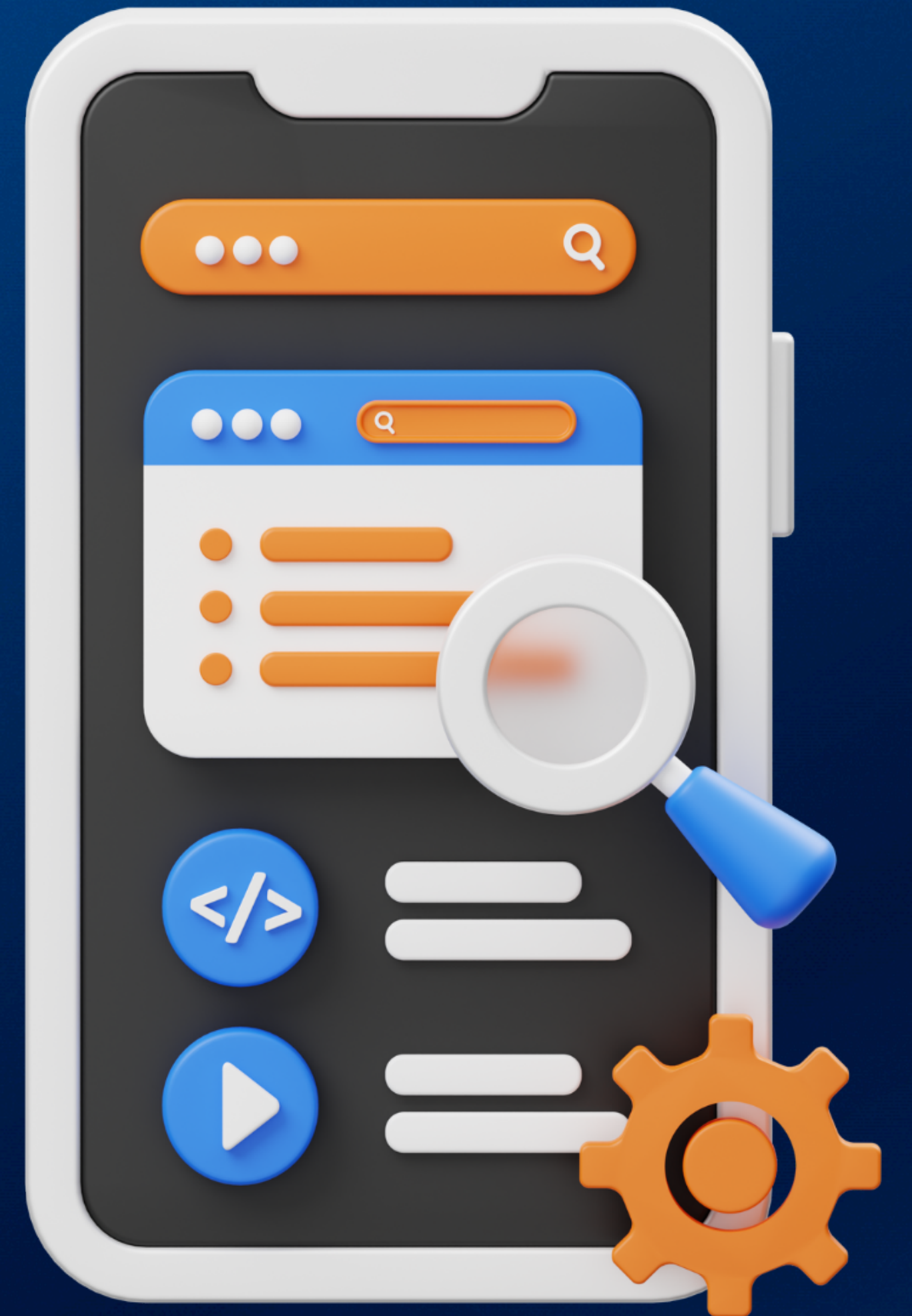
Software Developer @ClipMyHorse.TV

- Started as Web React developer
- 6+ years shipping React Native apps
- Co-founder of @FrontendQueens

# The promise vs. Reality



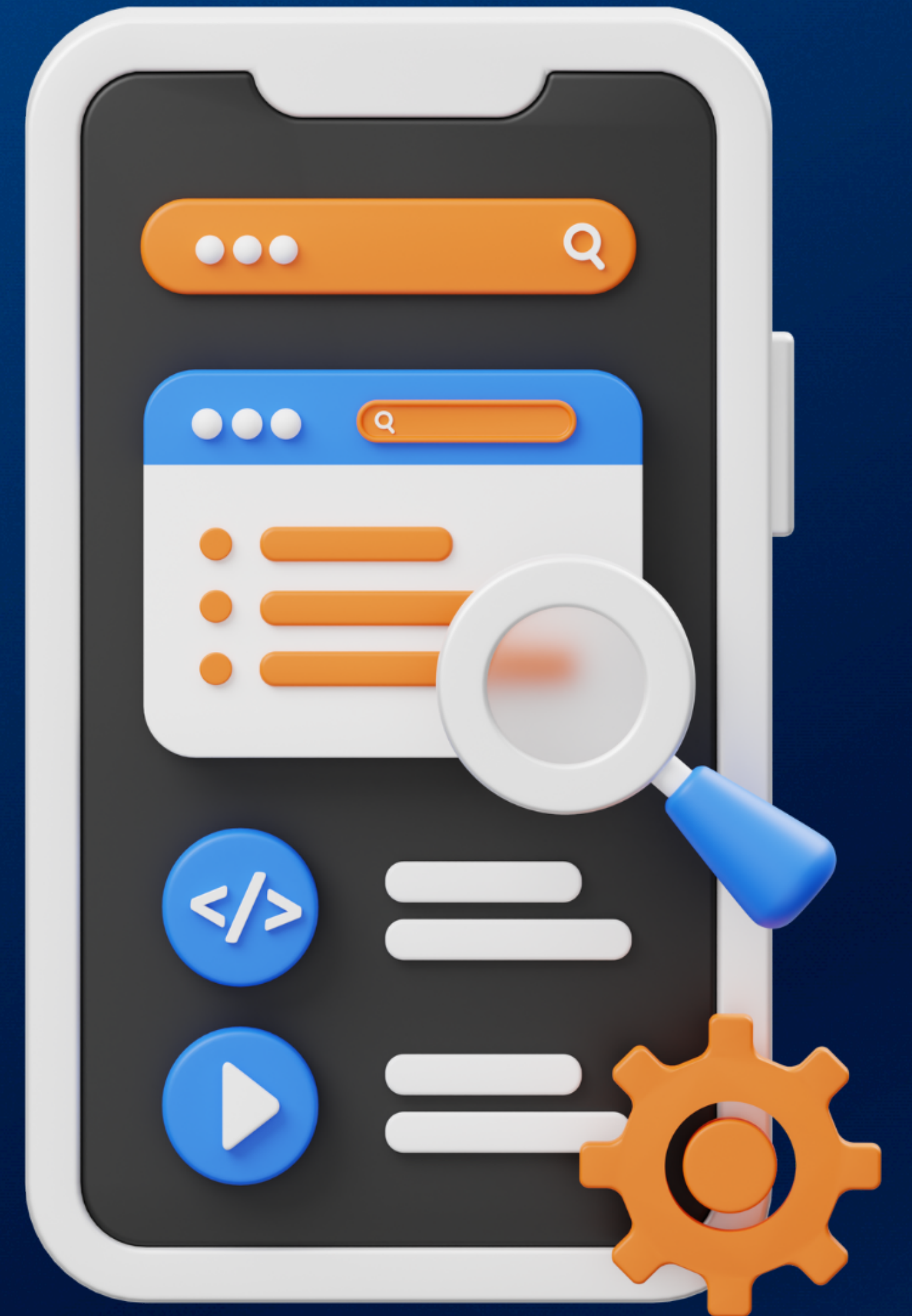
! = =



# The promise vs. Reality



! = =



**Why React Native Was Born?**

# The mobile-first imperative

2012

# The mobile-first imperative

2012

Facebook goes mobile-first

# The mobile-first imperative

2012

Facebook goes mobile-first



# The challenge



Web app inside  
a WebView

# The challenge



Web app inside  
a WebView



**A 2-day hackathon that  
changed everything**

# A 2-day hackathon that changed everything

mobile UI  
elements from  
JavaScript

# A 2-day hackathon that changed everything

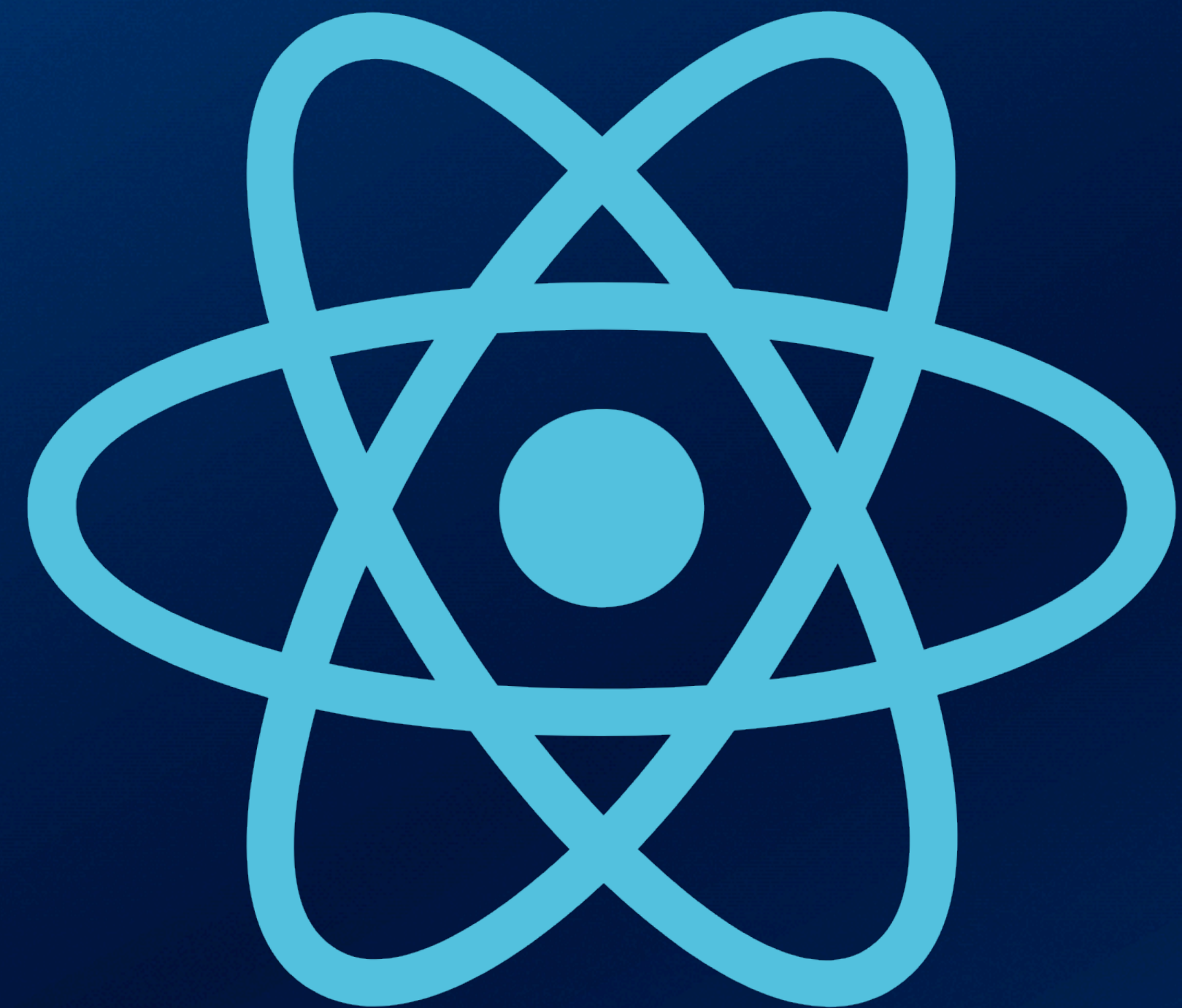
mobile UI  
elements from  
JavaScript



# 2015

React Conf

LEARN ONCE, WRITE  
ANYWHERE



**WHAT IS A MOBILE  
APPLICATION?**

# IT'S A **SIGNED** BUNDLE

- Binaries
- Assets
- Metadata

# IT'S A **SIGNED** BUNDLE

- Binaries
- Assets
- Metadata



# IT'S A **SIGNED** BUNDLE

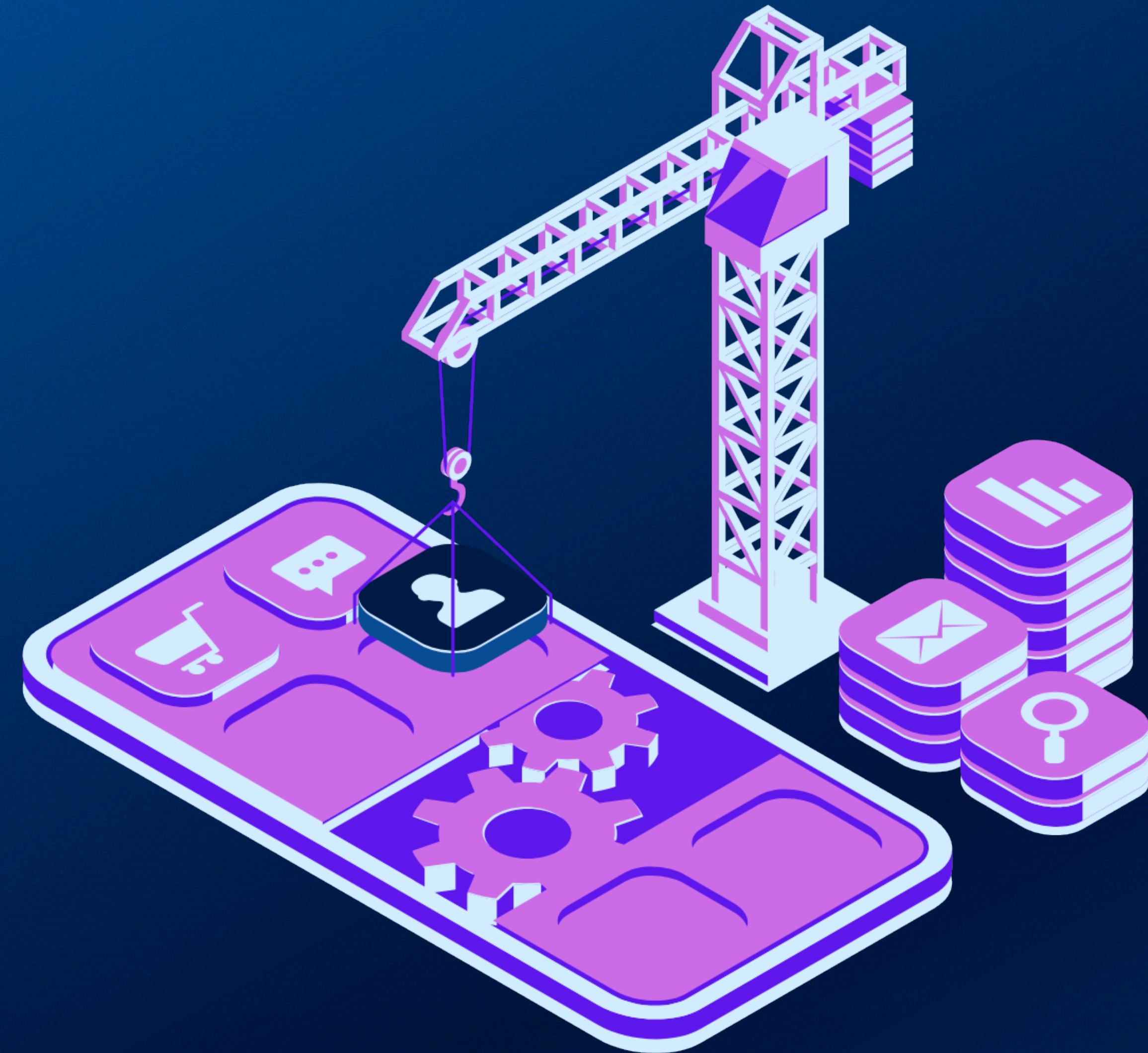
- Binaries
- Assets
- Metadata



.ipa (iOS)

.apk (Android)

# UNDER THE HOOD: THE NATIVE RUNTIME



# How JavaScript Becomes a Native App?

# How JavaScript Becomes a Native App?



JS Thread

# How JavaScript Becomes a Native App?



JS Thread



Native Thread

# How JavaScript Becomes a Native App?



JS Thread

The Bridge  
Asynchronous

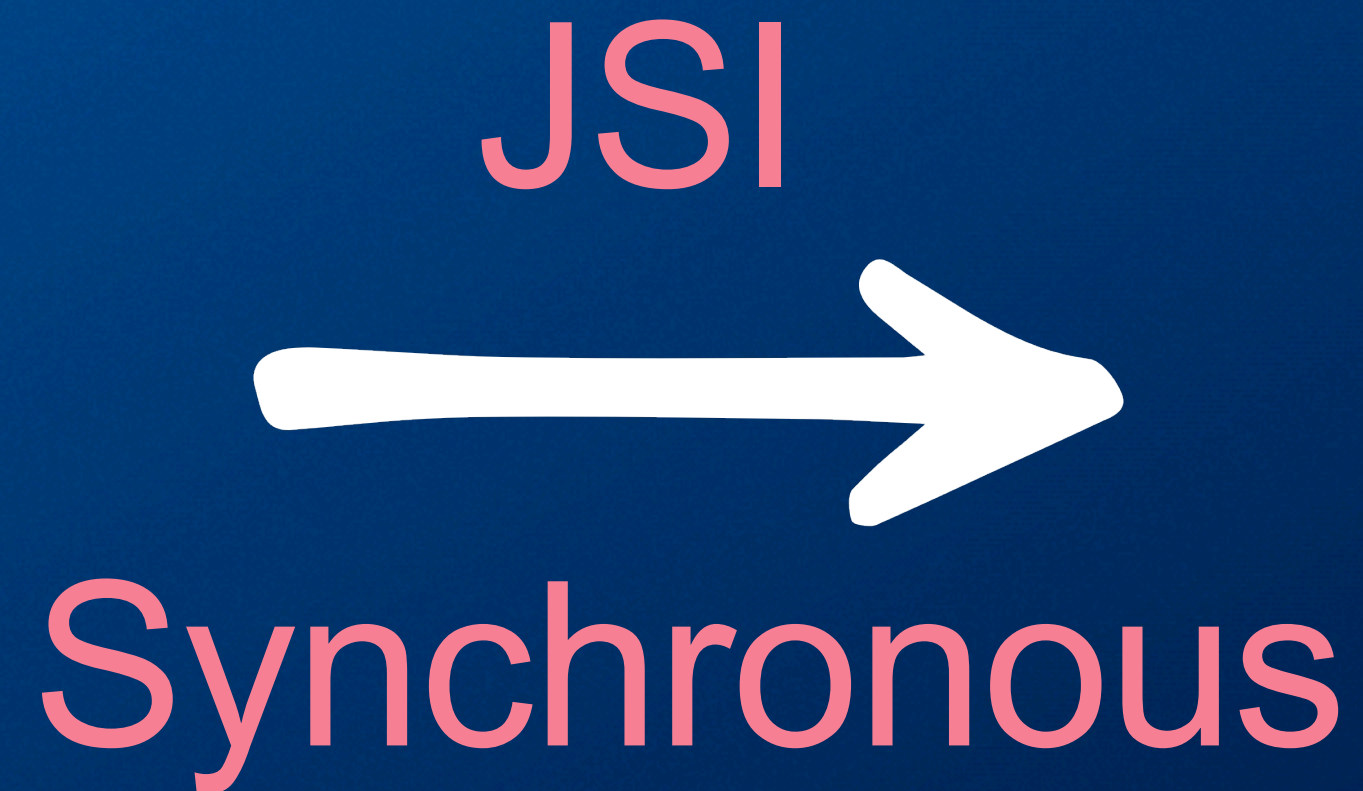
A pink dashed arrow points from the JavaScript logo area towards the native platform icons. The arrow is composed of several short horizontal dashes and ends in a solid pink arrowhead pointing to the right.

Native Thread

# The New Architecture



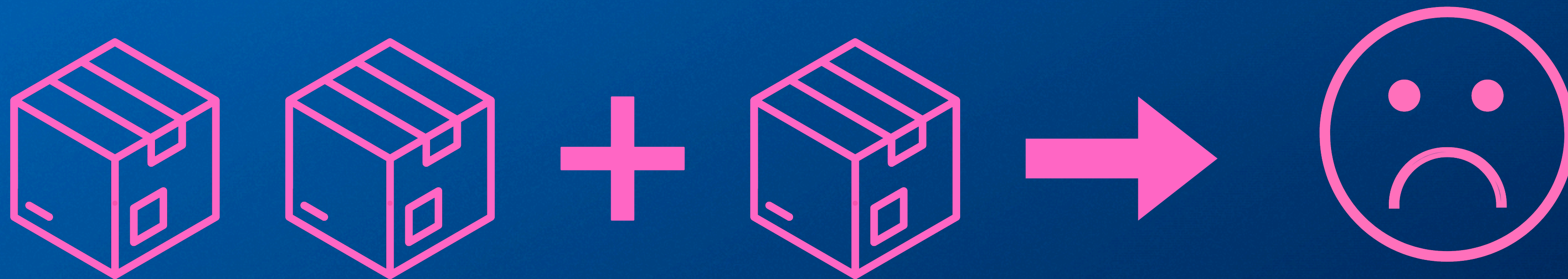
JS Thread



Native Thread

# Turbomodules

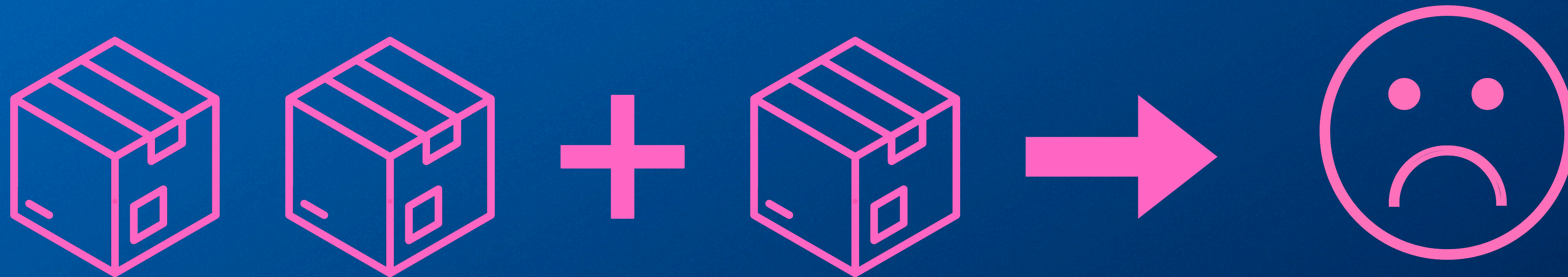
Before



Slow  
Startup

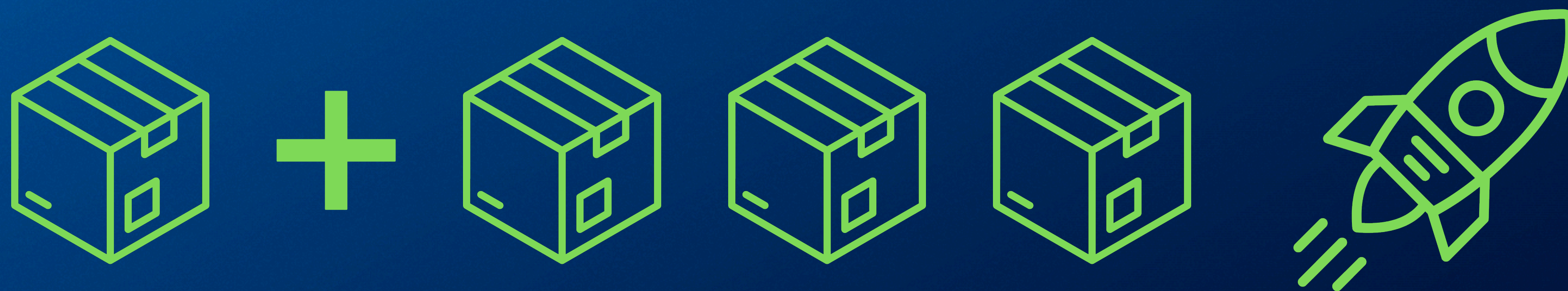
# Turbomodules

Before



Slow  
Startup

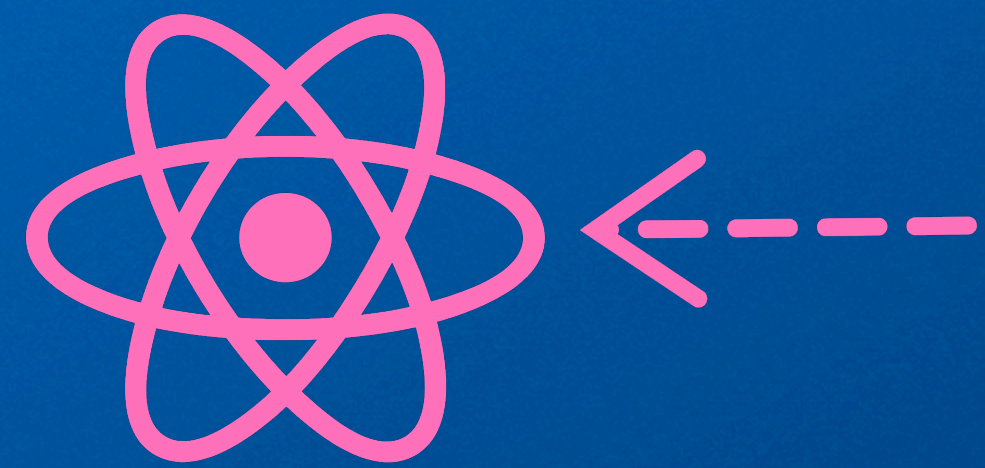
After



Faster  
Startup

# Fabric: The New Renderer

Before



Asynchronous  
Bridge



Slow  
Native UI

# Fabric: The New Renderer

Before



After



**From message passing →  
Direct communication**



# Components & Navigation

# Not your HTML



```
// Web  
<div style={{ backgroundColor: 'blue' }}>Hello</div>
```



```
// React Native  
<View style={{ backgroundColor: 'blue' }}>  
  <Text>Hello</Text>  
</View>
```

# Not your HTML



```
// Web  
<div style={{ backgroundColor: 'blue' }}>Hello</div>
```



```
// React Native  
<View style={{ backgroundColor: 'blue' }}>  
  <Text>Hello</Text>  
</View>
```

# Not your HTML

- `<View>` becomes a native `android.view.View`

on Android or a `UIView` on iOS

- `<Text>` becomes `TextView` on Android or

`UILabel` on iOS

- `<TextInput>` replaces regular `<input>`

# Native Frameworks



Android Jetpack



UIKit

# Styling

## Web

```
// Web
.container {
  display: flex;
  flex-direction: row; // Default in web
  justify-content: center;
  align-items: center;
  background-color: #f0f0f0;
  box-shadow: 2px 2px 5px rgba(0,0,0,0.2);
  border-radius: 10px;
  padding: 20px;
  width: 300px;
  height: 200px;
}

.text {
  color: #333;
  font-size: 16px;
  font-family: Arial, sans-serif;
}
```

## React Native

```
//React Native
const styles = StyleSheet.create({
  container: {
    flexDirection: 'column', // Default in React Native
    justifyContent: 'center',
    alignItems: 'center',
    backgroundColor: '#f0f0f0',
    shadowColor: '#000', // Different shadow properties
    shadowOffset: { width: 2, height: 2 },
    shadowOpacity: 0.2,
    shadowRadius: 5,
    borderRadius: 10, // Single property
    padding: 20,
    width: 300,
    height: 200,
  },
  text: {
    color: '#333',
    fontSize: 16,
    fontFamily: 'Arial', // Limited font support, may require linking
  },
});
```

# Styling

## Web

```
// Web
.container {
  display: flex;
  flex-direction: row; // Default in web
  justify-content: center;
  align-items: center;
  background-color: #f0f0f0;
  box-shadow: 2px 2px 5px rgba(0,0,0,0.2);
  border-radius: 10px;
  padding: 20px;
  width: 300px;
  height: 200px;
}

.text {
  color: #333;
  font-size: 16px;
  font-family: Arial, sans-serif;
}
```

## React Native

```
//React Native
const styles = StyleSheet.create({
  container: {
    flexDirection: 'column', // Default in React Native
    justifyContent: 'center',
    alignItems: 'center',
    backgroundColor: '#f0f0f0',
    shadowColor: '#000', // Different shadow properties
    shadowOffset: { width: 2, height: 2 },
    shadowOpacity: 0.2,
    shadowRadius: 5,
    borderRadius: 10, // Single property
    padding: 20,
    width: 300,
    height: 200,
  },
  text: {
    color: '#333',
    fontSize: 16,
    fontFamily: 'Arial', // Limited font support, may require linking
  },
});
```

# Styling

## Web

```
// Web
.container {
  display: flex;
  flex-direction: row; // Default in web
  justify-content: center;
  align-items: center;
  background-color: #f0f0f0;
  box-shadow: 2px 2px 5px rgba(0,0,0,0.2);
  border-radius: 10px;
  padding: 20px;
  width: 300px;
  height: 200px;
}

.text {
  color: #333;
  font-size: 16px;
  font-family: Arial, sans-serif;
}
```

## React Native

```
//React Native
const styles = StyleSheet.create({
  container: {
    flexDirection: 'column', // Default in React Native
    justifyContent: 'center',
    alignItems: 'center',
    backgroundColor: '#f0f0f0',
    shadowColor: '#000', // Different shadow properties
    shadowOffset: { width: 2, height: 2 },
    shadowOpacity: 0.2,
    shadowRadius: 5,
    borderRadius: 10, // Single property
    padding: 20,
    width: 300,
    height: 200,
  },
  text: {
    color: '#333',
    fontSize: 16,
    fontFamily: 'Arial', // Limited font support, may require linking
  },
});
```

# Styling

## Web

```
// Web
.container {
  display: flex;
  flex-direction: row; // Default in web
  justify-content: center;
  align-items: center;
  background-color: #f0f0f0;
  box-shadow: 2px 2px 5px rgba(0,0,0,0.2);
  border-radius: 10px;
  padding: 20px;
  width: 300px;
  height: 200px;
}

.text {
  color: #333;
  font-size: 16px;
  font-family: Arial, sans-serif;
}
```

## React Native

```
//React Native
const styles = StyleSheet.create({
  container: {
    flexDirection: 'column', // Default in React Native
    justifyContent: 'center',
    alignItems: 'center',
    backgroundColor: '#f0f0f0',
    shadowColor: '#000', // Different shadow properties
    shadowOffset: { width: 2, height: 2 },
    shadowOpacity: 0.2,
    shadowRadius: 5,
    borderRadius: 10, // Single property
    padding: 20,
    width: 300,
    height: 200,
  },
  text: {
    color: '#333',
    fontSize: 16,
    fontFamily: 'Arial', // Limited font support, may require linking
  },
});
```

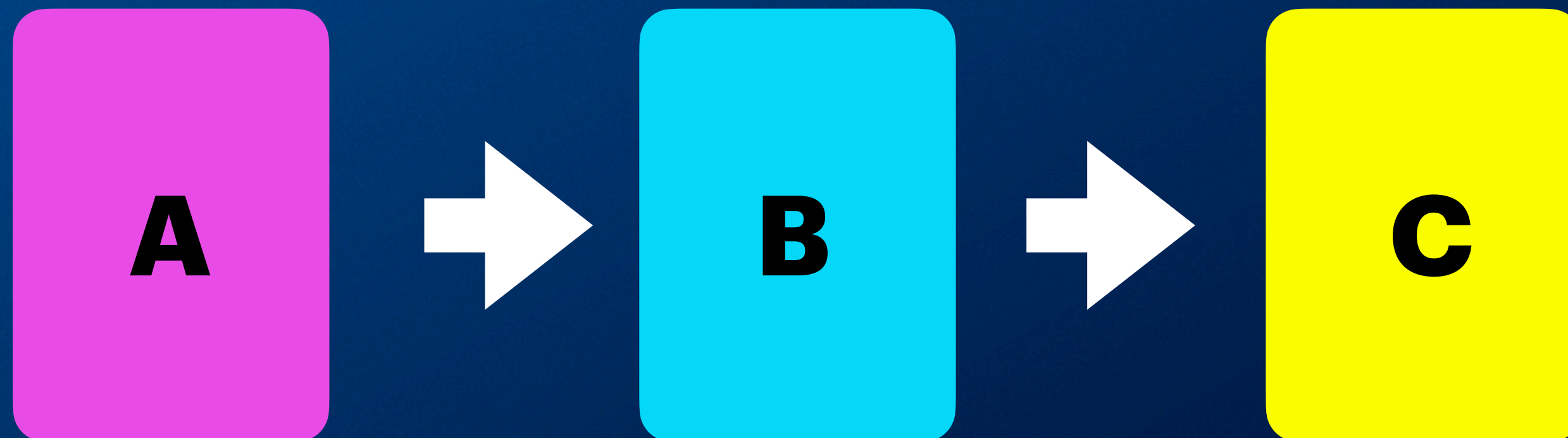
# Navigation - The Stack is Real



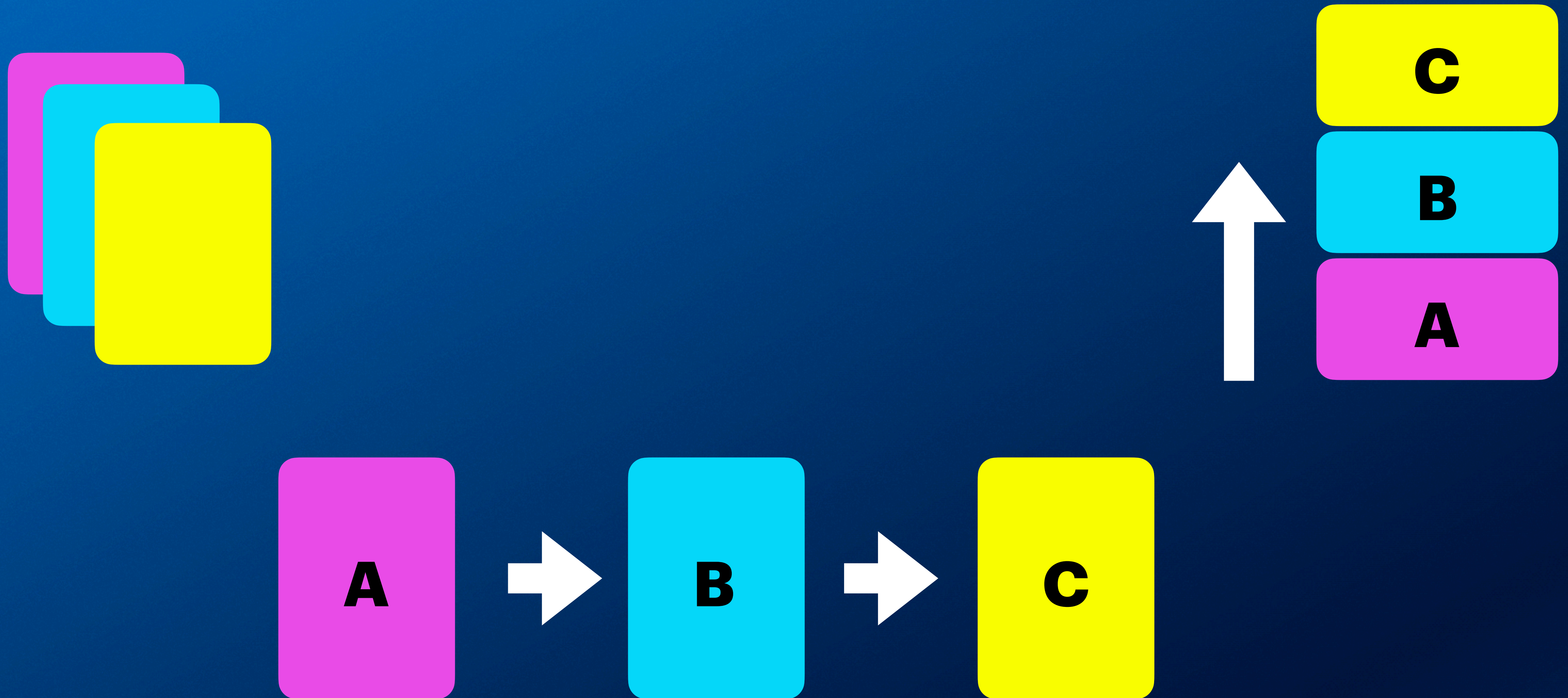
# Navigation - The Stack is Real



# Navigation - The Stack is Real



# Navigation - The Stack is Real



# Navigation - The Stack is Real



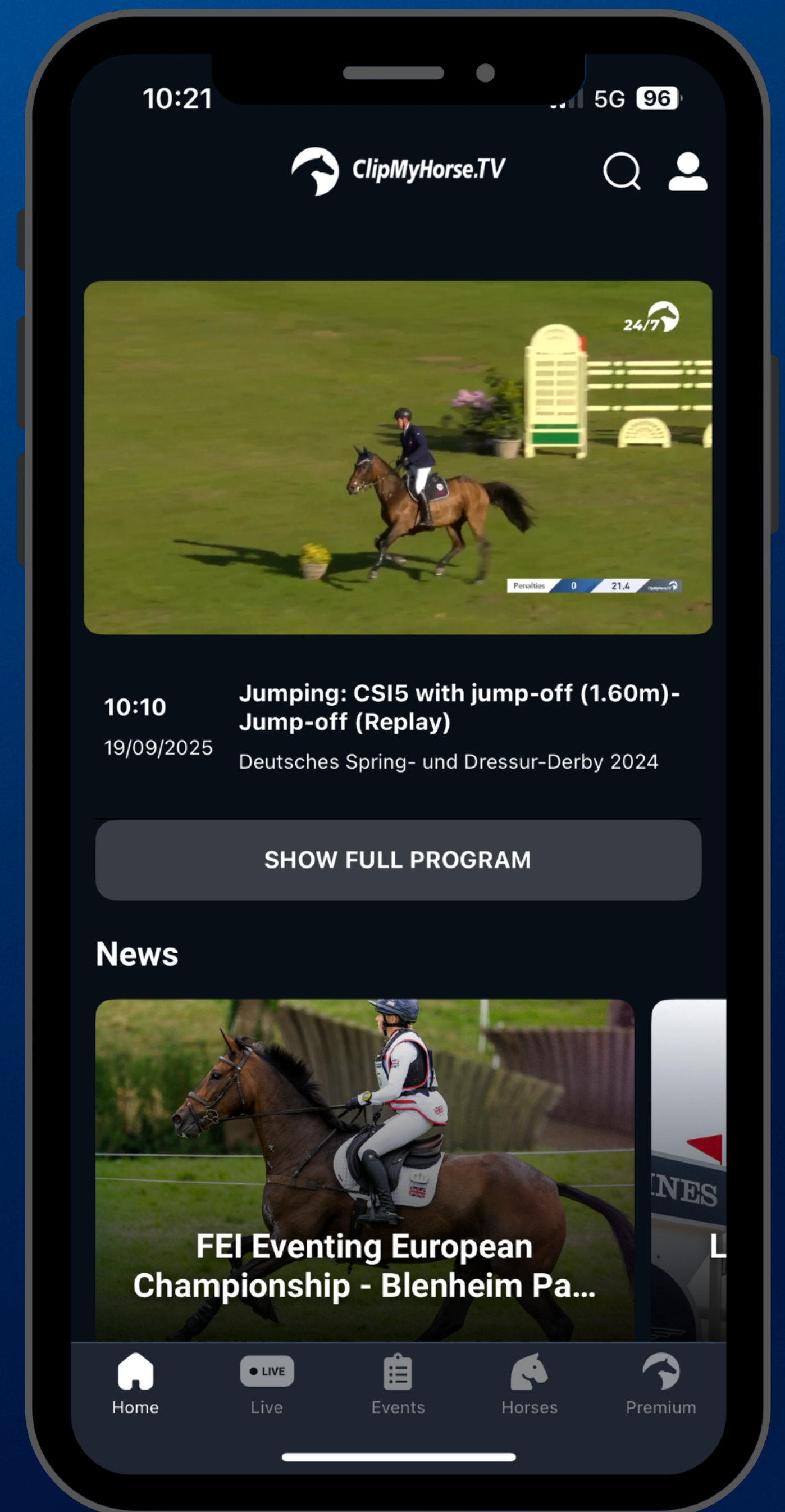
React Navigation



Expo Router

```
const Tab = createBottomTabNavigator();
```

```
const App = () => {  
  return (  
    <NavigationContainer>  
      <Tab.Navigator>  
        <Tab.Screen name="Home" component={HomeScreen} />  
        <Tab.Screen name="Live" component={LiveScreen} />  
        <Tab.Screen name="Events" component={EventsScreen} />  
        <Tab.Screen name="Horses" component={HorsesScreen} />  
        <Tab.Screen name="Premium" component={PremiumScreen} />  
      </Tab.Navigator>  
    </NavigationContainer>  
  );  
};
```



```
const Tab = createBottomTabNavigator();
```

```
const App = () => {  
  return (  
    <NavigationContainer>
```

```
      <Tab.Navigator>
```

```
        <Tab.Screen name="Home" component={HomeScreen} />
```

```
        <Tab.Screen name="Live" component={LiveScreen} />
```

```
        <Tab.Screen name="Events" component={EventsScreen} />
```

```
        <Tab.Screen name="Horses" component={HorsesScreen} />
```

```
        <Tab.Screen name="Premium" component={PremiumScreen} />
```

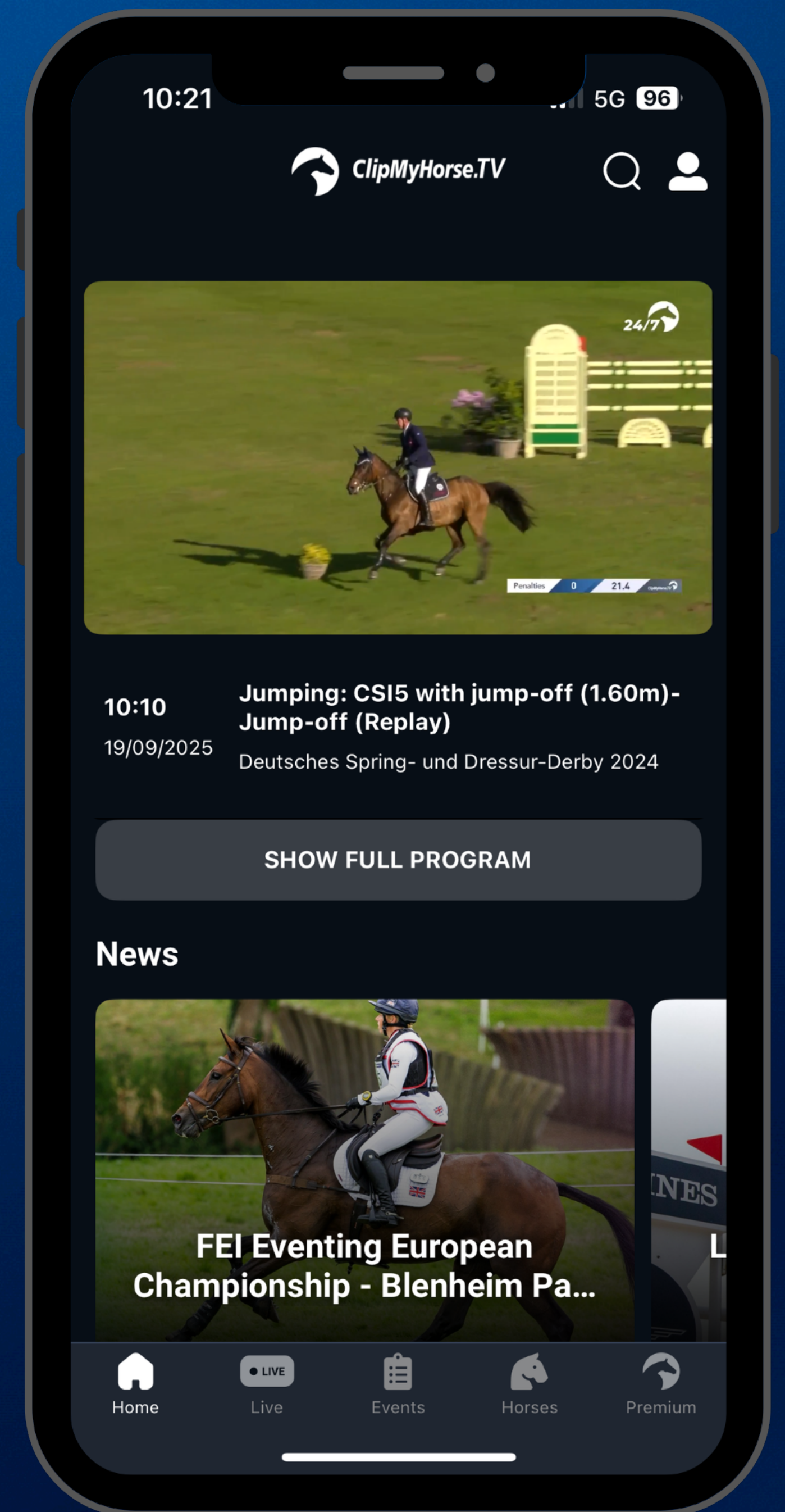
```
      </Tab.Navigator>
```

```
    </NavigationContainer>
```

```
  );  
</NavigationContainer>
```

```
);
```

```
};
```



```
const Tab = createBottomTabNavigator();
```

```
const App = () => {  
  return (  
    <NavigationContainer>  
      <Tab.Navigator>
```

```
        <Tab.Screen name="Home" component={HomeScreen} />
```

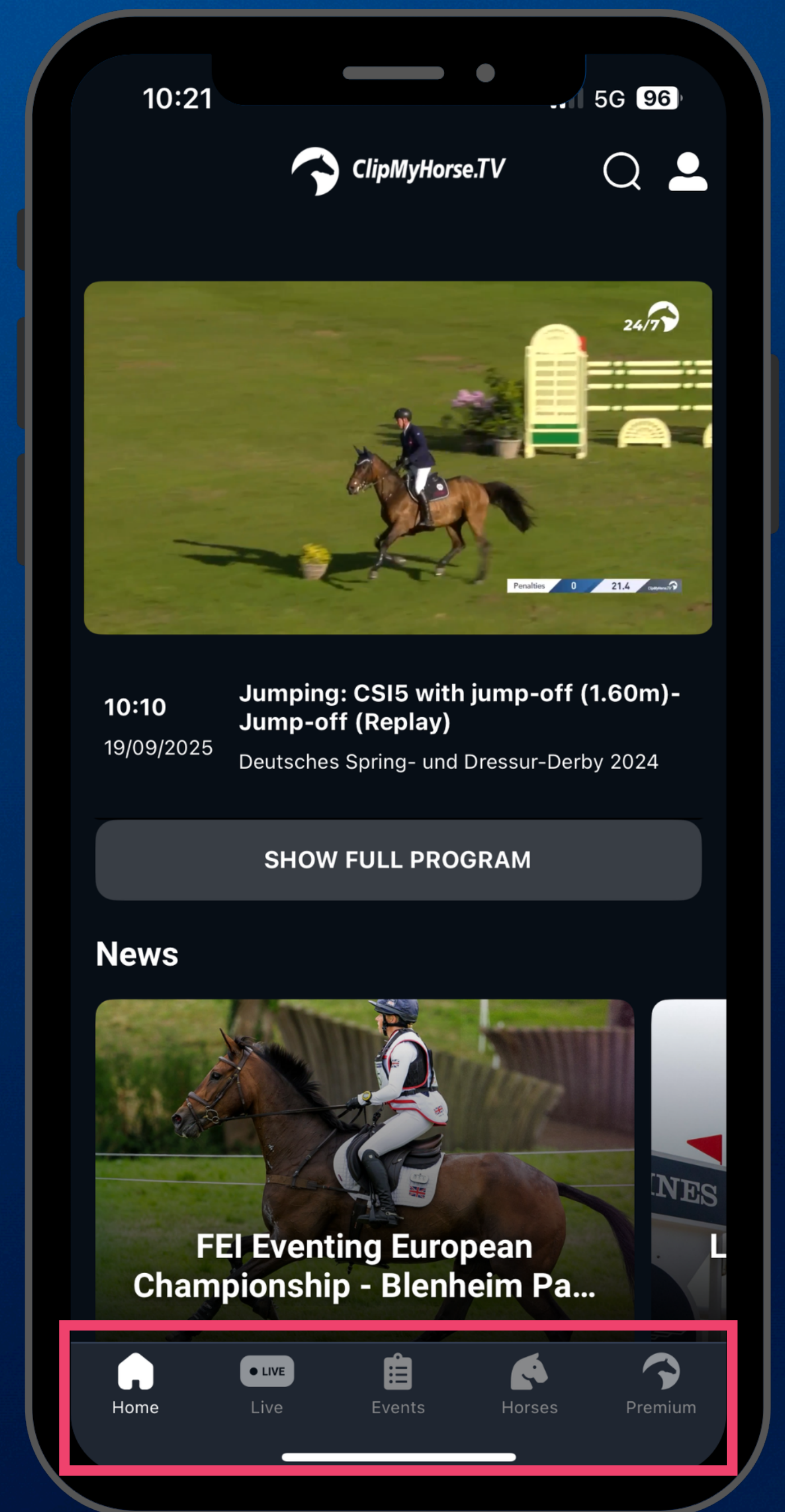
```
        <Tab.Screen name="Live" component={LiveScreen} />
```

```
        <Tab.Screen name="Events" component={EventsScreen} />
```

```
        <Tab.Screen name="Horses" component={HorsesScreen} />
```

```
        <Tab.Screen name="Premium" component={PremiumScreen} />
```

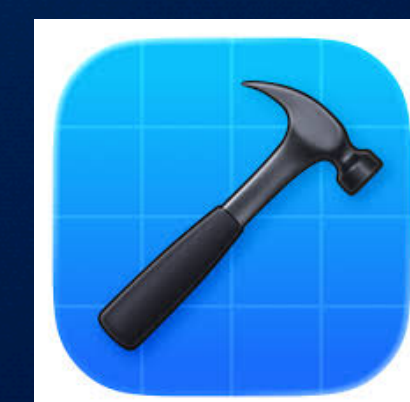
```
      </Tab.Navigator>  
    </NavigationContainer>  
  );  
};
```



# Debugging & Tooling

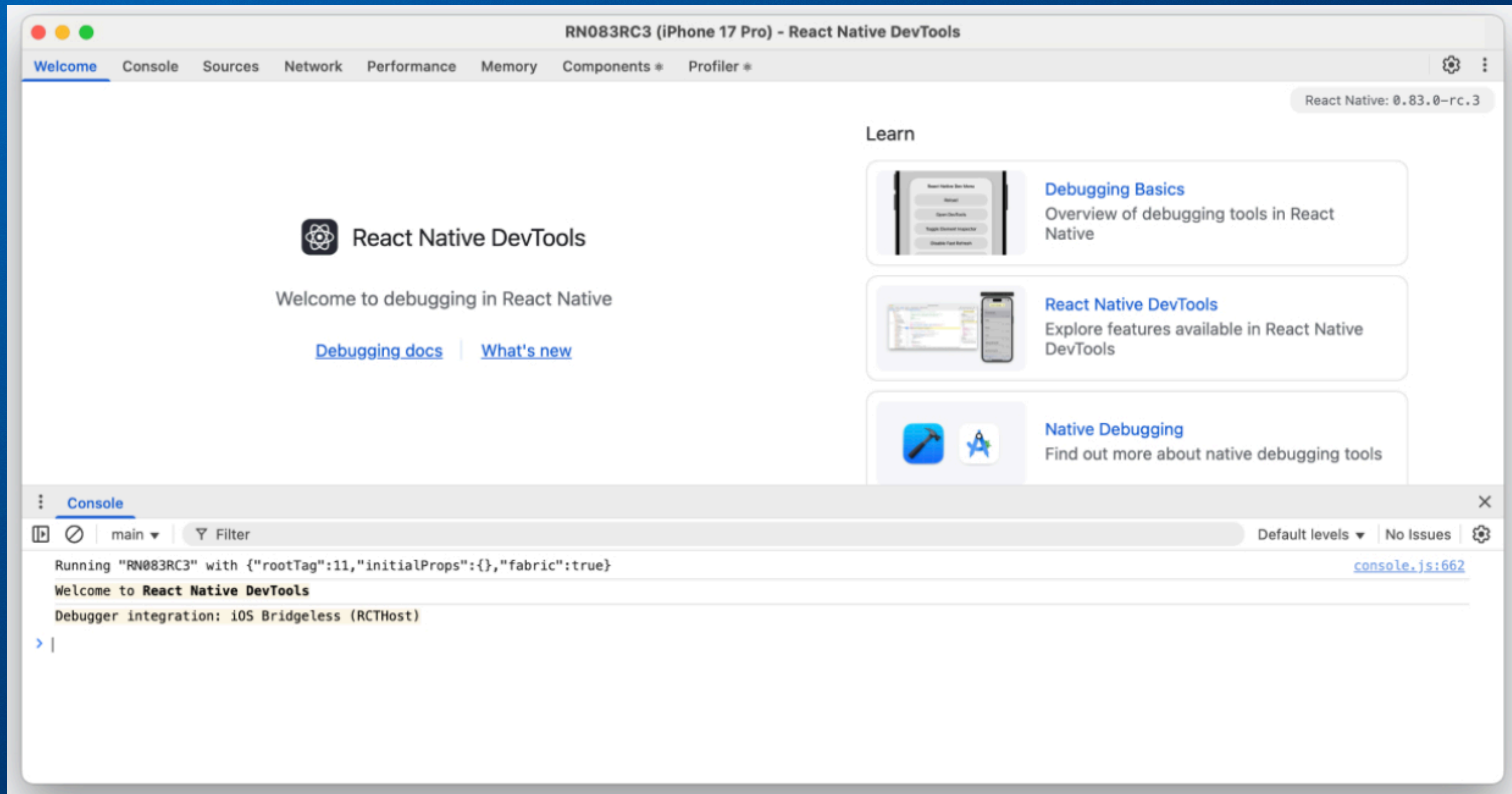


Metro



Flipper

# Debugging & Tooling



# Debugging & Tooling

The screenshot shows the React Native DevTools interface for an iPhone 17 Pro simulator. The window title is "RN083RC3 (iPhone 17 Pro) - React Native DevTools". The top navigation bar includes "Welcome", "Console", "Sources", "Network", "Performance", "Memory", "Components", and "Profiler". The "Welcome" tab is active, displaying the React Native DevTools logo and the text "Welcome to debugging in React Native". Below this, there are links for "Debugging docs" and "What's new". On the right side, there is a "Learn" section with three cards: "Debugging Basics" (Overview of debugging tools in React Native), "React Native DevTools" (Explore features available in React Native DevTools), and "Native Debugging" (Find out more about native debugging tools). The bottom panel shows the "Console" tab with the following log messages: "Running 'RN083RC3' with {\"rootTag\":11,\"initialProps\":{},\"fabric\":true}", "Welcome to React Native DevTools", and "Debugger integration: iOS Bridgeless (RCTHost)". The console also shows a cursor at the end of the line. The top right corner of the console panel indicates "Default levels" and "No Issues".

# My advice: start with Expo



Expo

- Expo Go app
- Expo Router
- EAS Build

**And the story doesn't stop at  
phones...**

# And the story doesn't stop at phones...

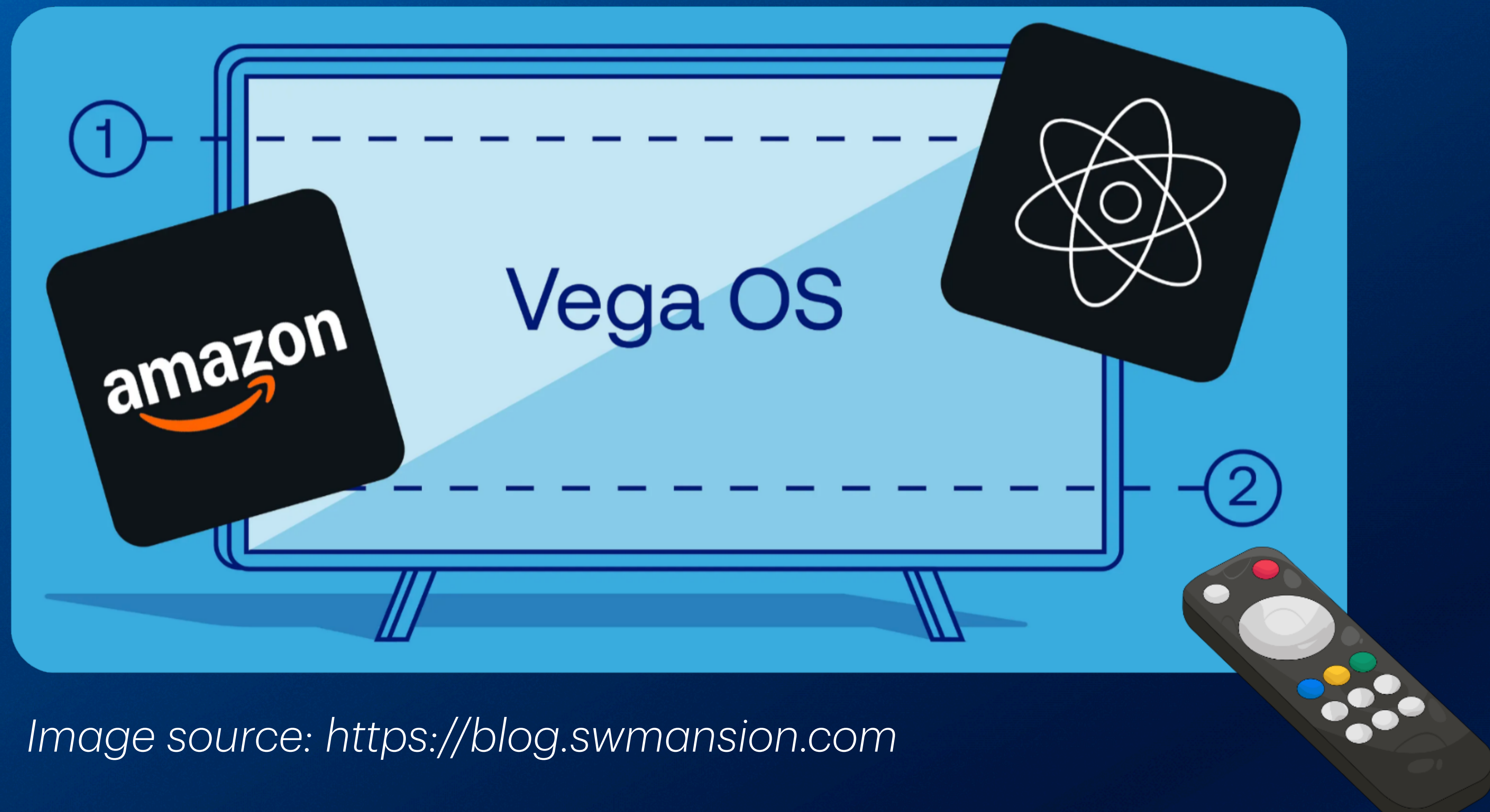


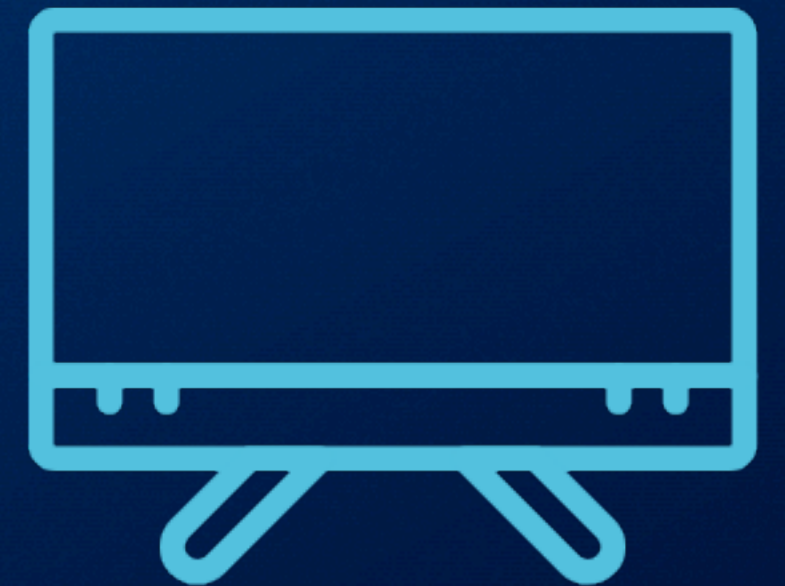
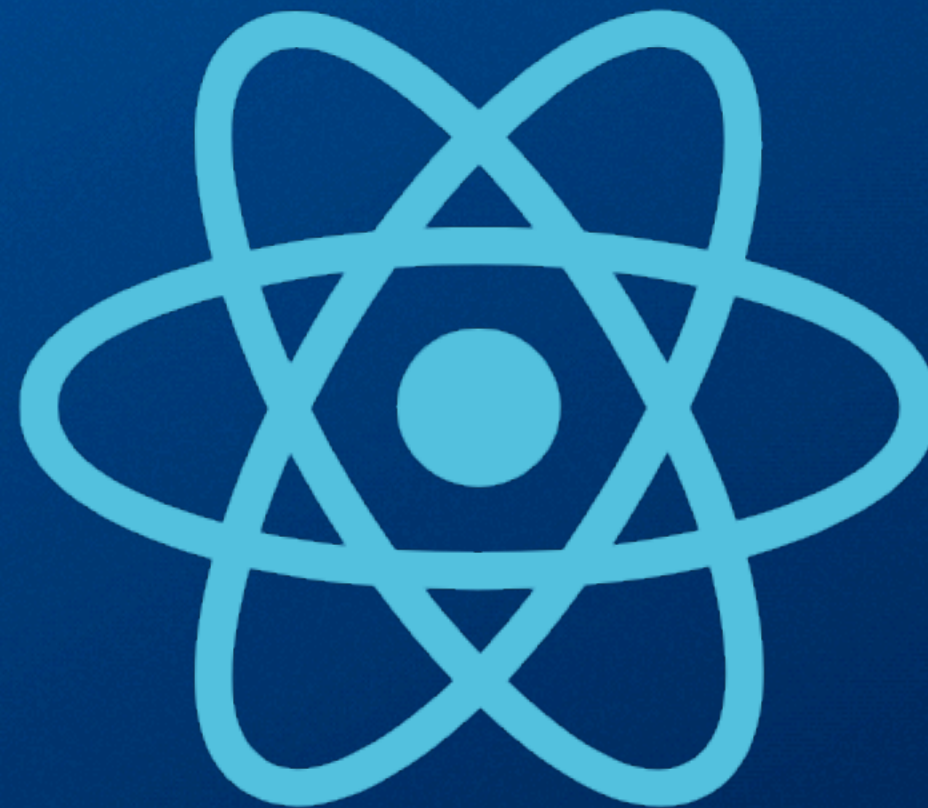
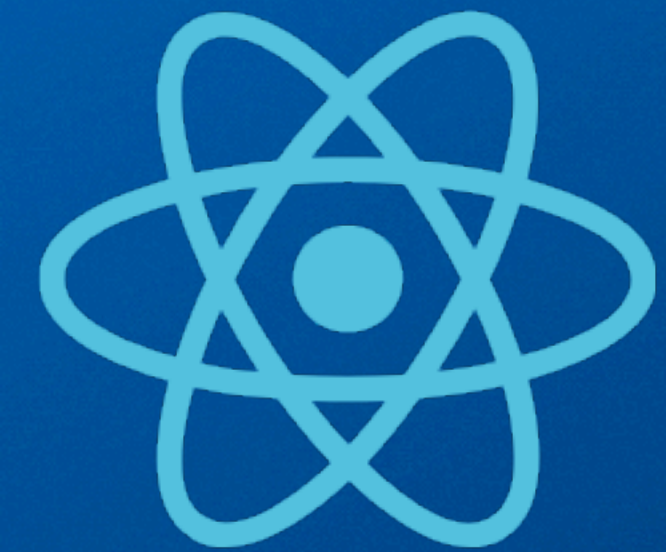
Image source: <https://blog.swmansion.com>

# The React Journey is Just Beginning

**Web React**

**React Native**

**Beyond Mobile**

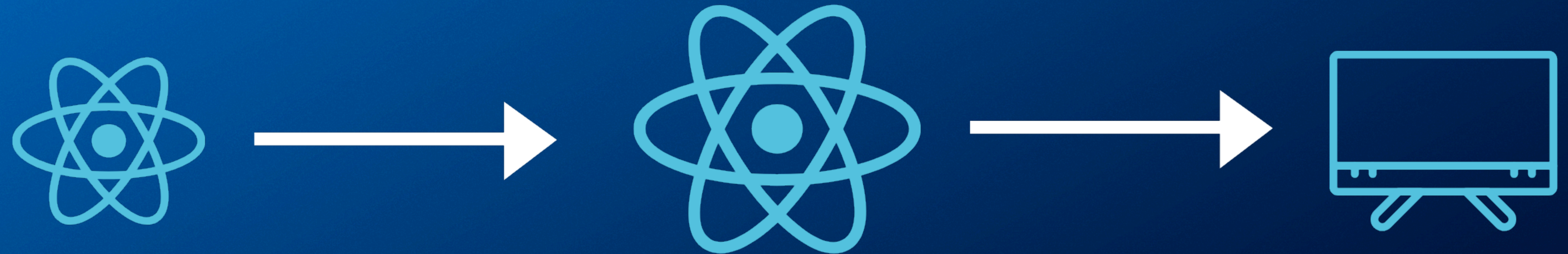


# The React Journey is Just Beginning

**Web React**

**React Native**

**Beyond Mobile**



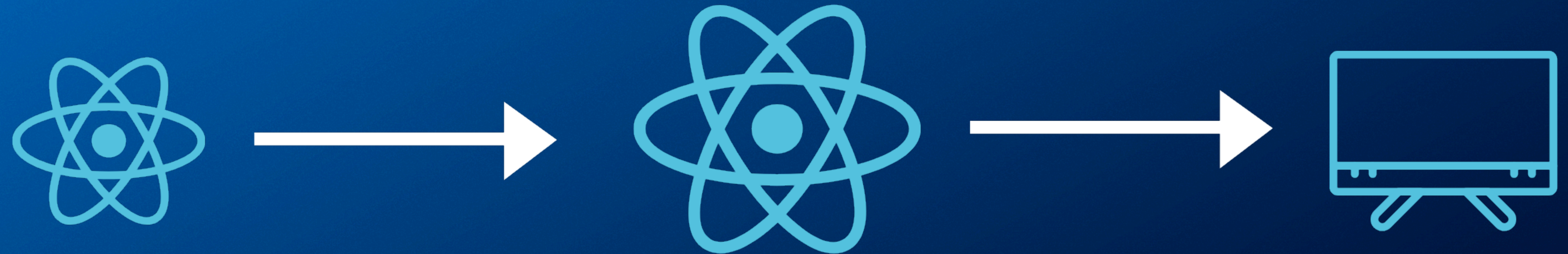
Learn Once, Write Anywhere

# The React Journey is Just Beginning

**Web React**

**React Native**

**Beyond Mobile**



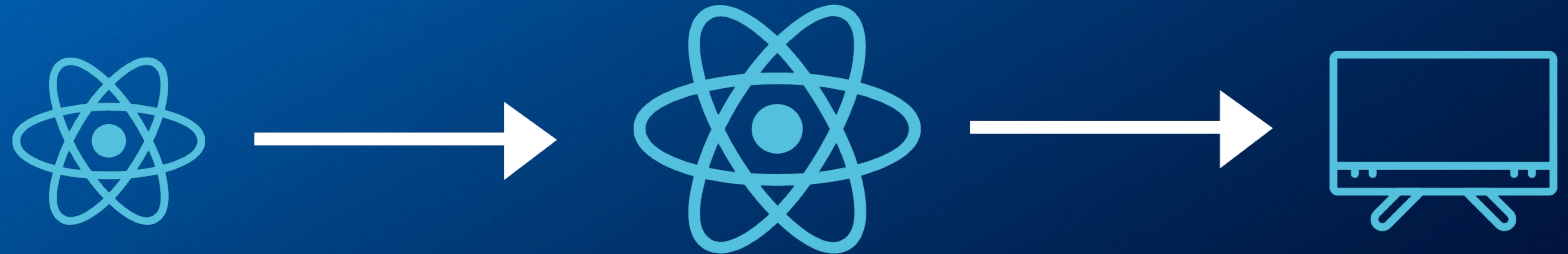
Learn Once, Write Anywhere

# The React Journey is Just Beginning

**Web React**

**React Native**

**Beyond Mobile**

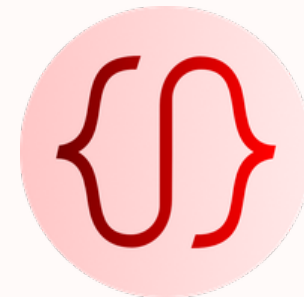


Learn Once, Write Anywhere

WEB  DAY

2026

Kudos



WEB  DAY

2026

**Thanks!**