

The background is a solid purple gradient. Overlaid on this are numerous abstract, colorful shapes. These include circles, elongated capsules, and thick diagonal stripes in shades of blue, green, yellow, orange, and red. The shapes are scattered across the frame, with some appearing more prominent than others.

# ReactNative

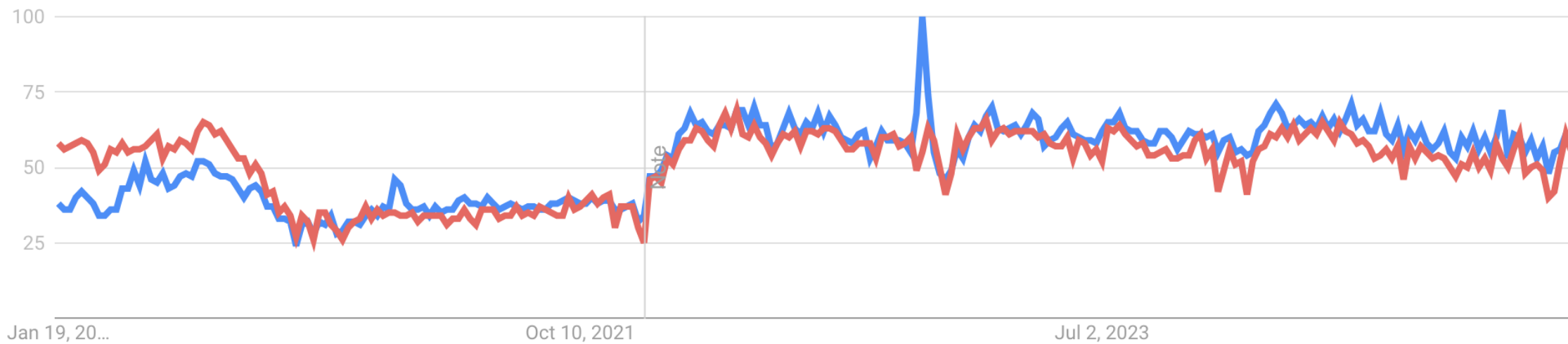
Why learning  
RN ?



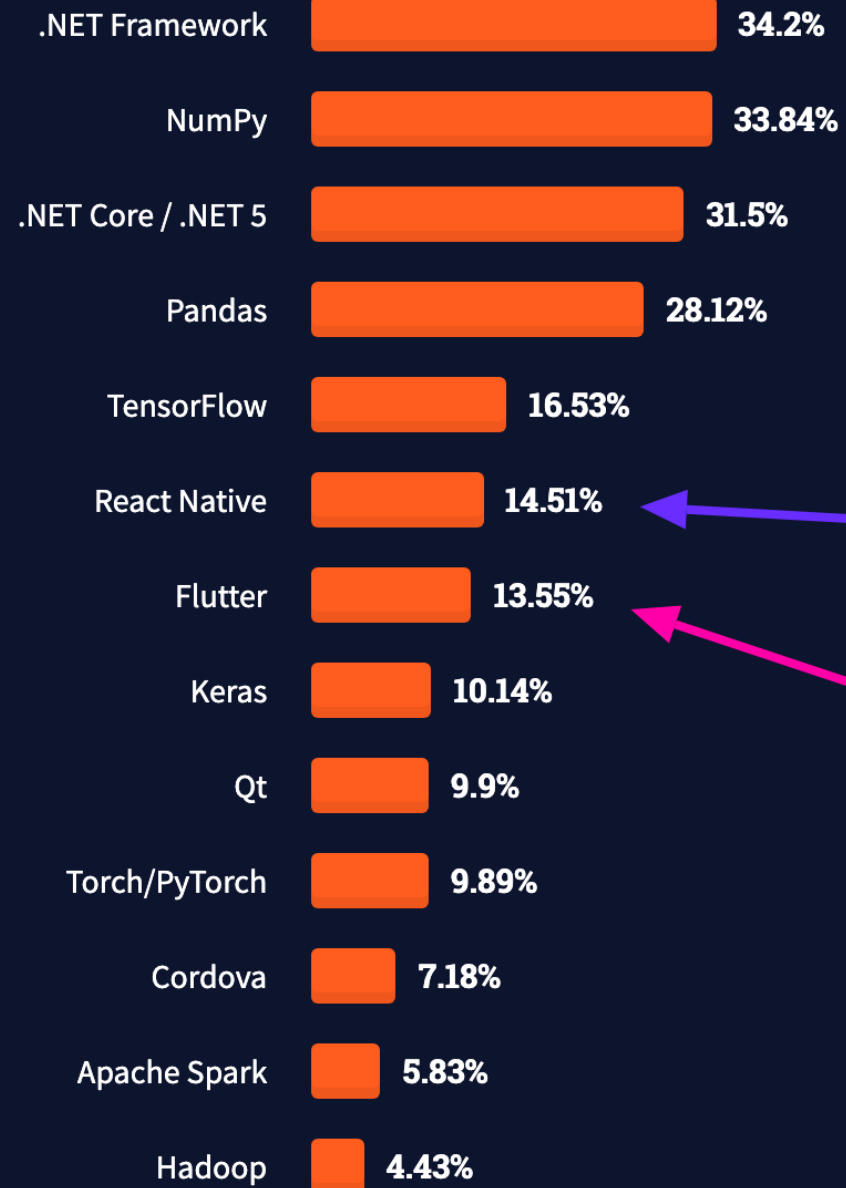
# Why learning RN ?

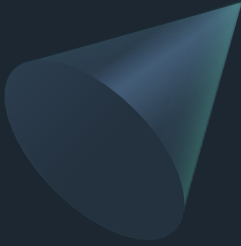


Average




# Why learning RN ?





# Why learning RN ?

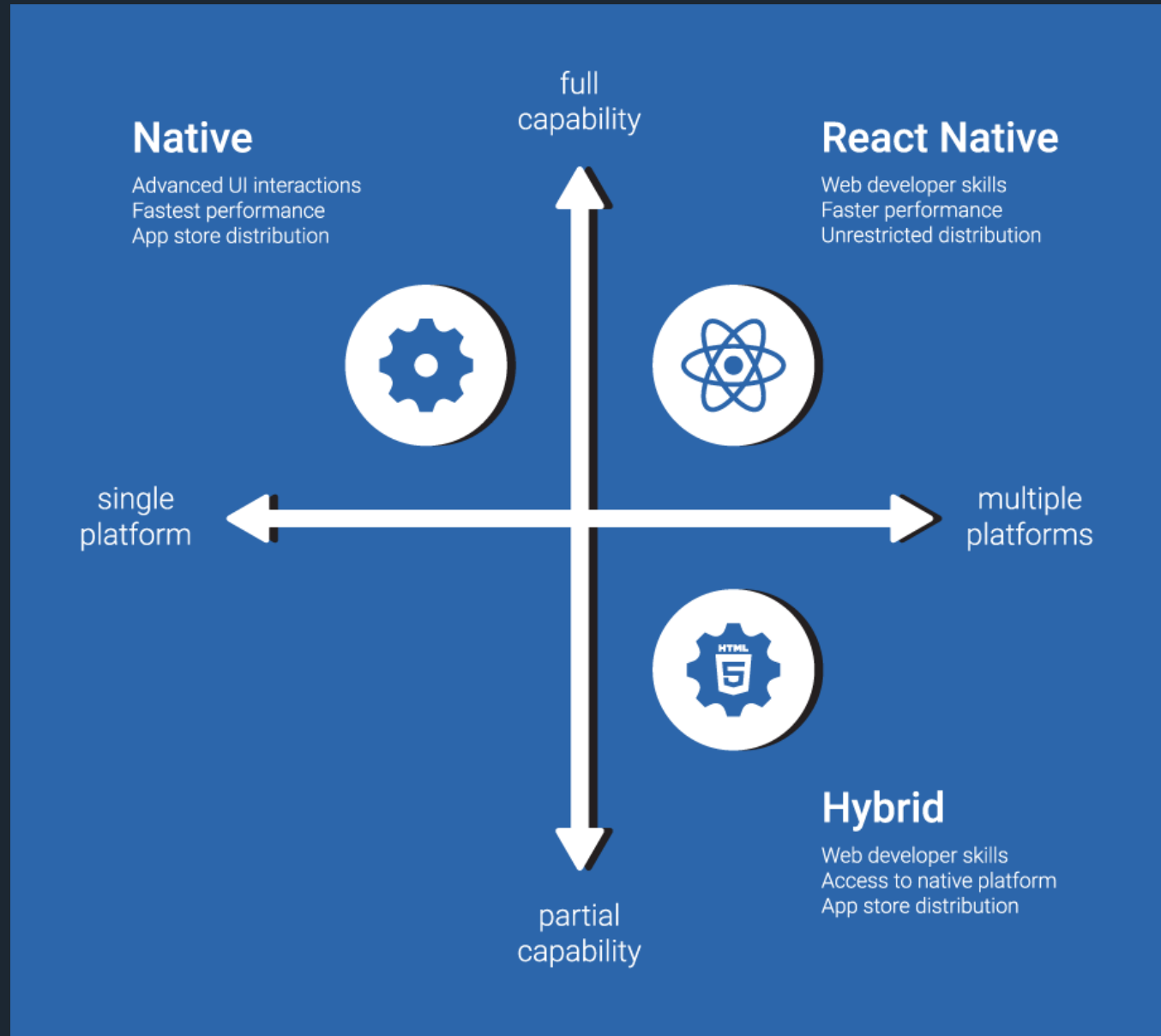
	Flutter	React Native
Stack Overflow questions	202,405	142,308
GitHub issues	13,184 (89,313 closed)	582 (25,965 closed)
GitHub forks	27.9K	24.4K
GitHub stars	168K	120K
Reddit members	147K	145K
Twitter followers	264.6K	213.9K



	Indeed	Glassdoor	SimplyHired
Flutter	1,029	554	262
React Native	4,997	2,291	1,222

*Data from 2025.*

# Why learning RN ?



# React JS

VS

# React Native

- 2011 (public version 2013)
- Web Apps (HTML)
- JS Libraries
- CSS

- 2013 (public version 2015)
- Cross-platform mobile apps
- Its own libraries
- Stylesheet



# React JS

# VS

# React Native

```
import React, { useState } from "react";

function App() {
  const [count, setCount] = useState(0);

  return (
    <div>
      <p>{count}</p>
      <button onClick={() => setCount(count + 1)}>+ 1</button>
    </div>
  );
}
```

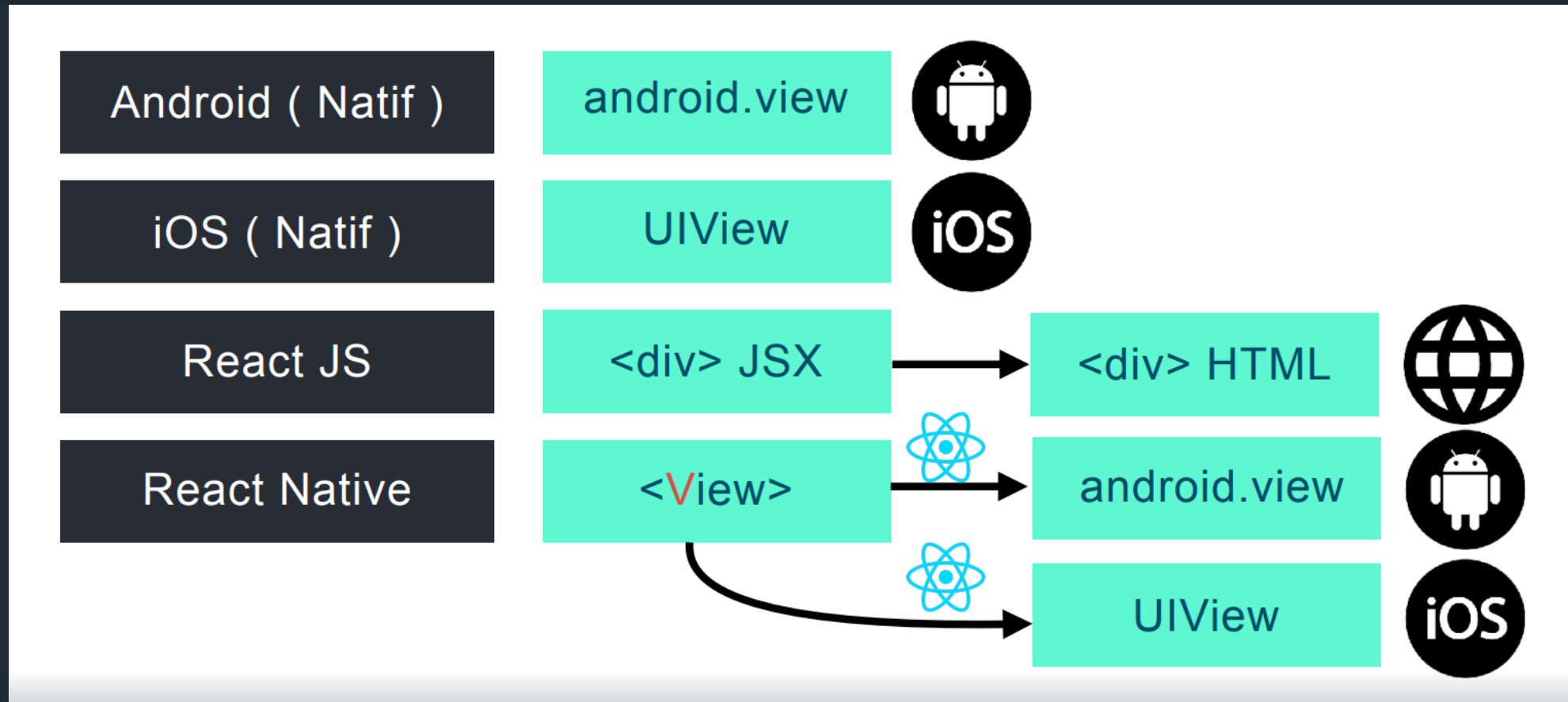
```
import React, { useState } from "react";

function App() {
  const [count, setCount] = useState(0);

  return (
    <View>
      <Text>{count}</Text>
      <Button onPress={() => setCount(count + 1)} title="+1" />
    </View>
  );
}
```



# Platforms differences



# Compiled or not ?

```
import React, { useState } from "react";
```

```
function App() {
```

```
  const [count, setCount] = useState(0);
```

```
  return (
```

```
    <View>
```

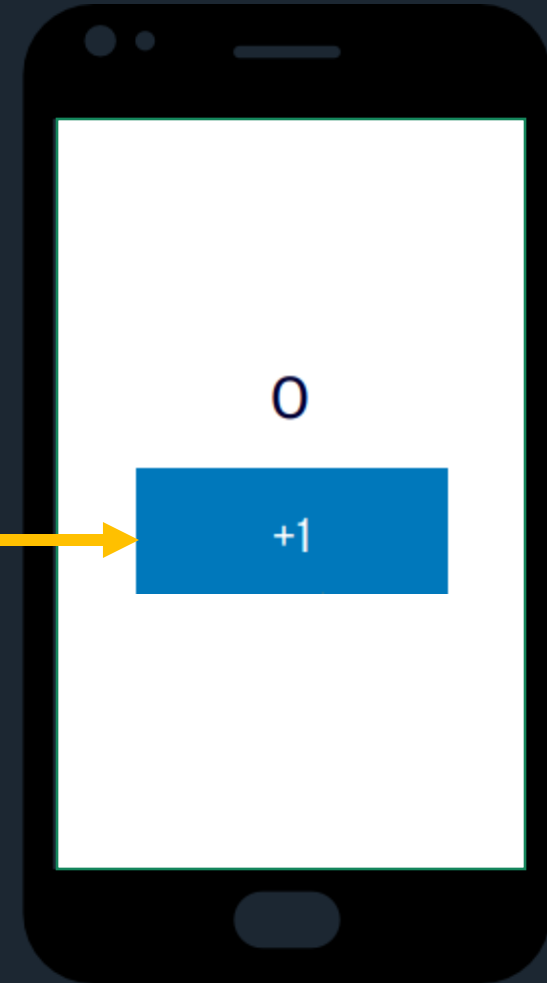
```
      <Text>{count}</Text>
```

```
      <Button onPress={() => setCount(count + 1)} title="+1" />
```

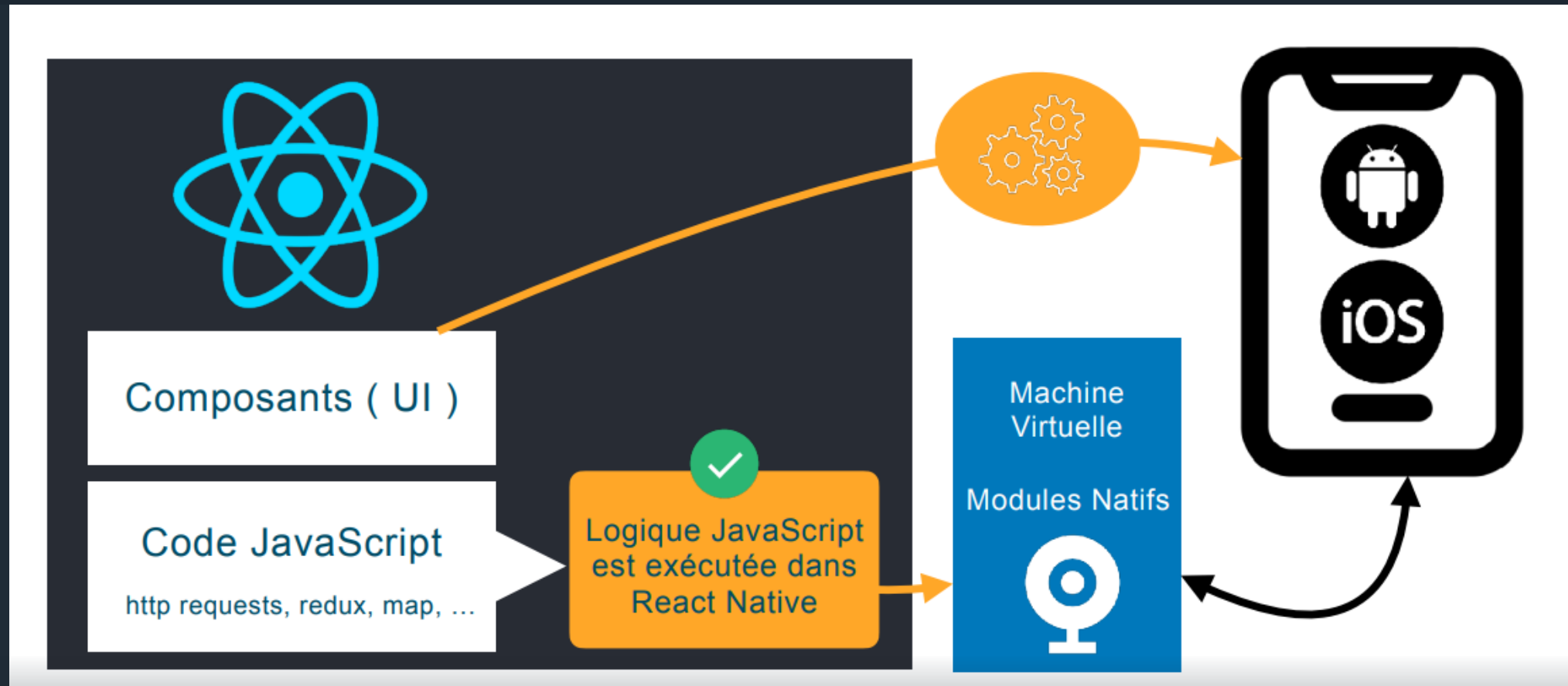
```
    </View>
```

```
  );
```

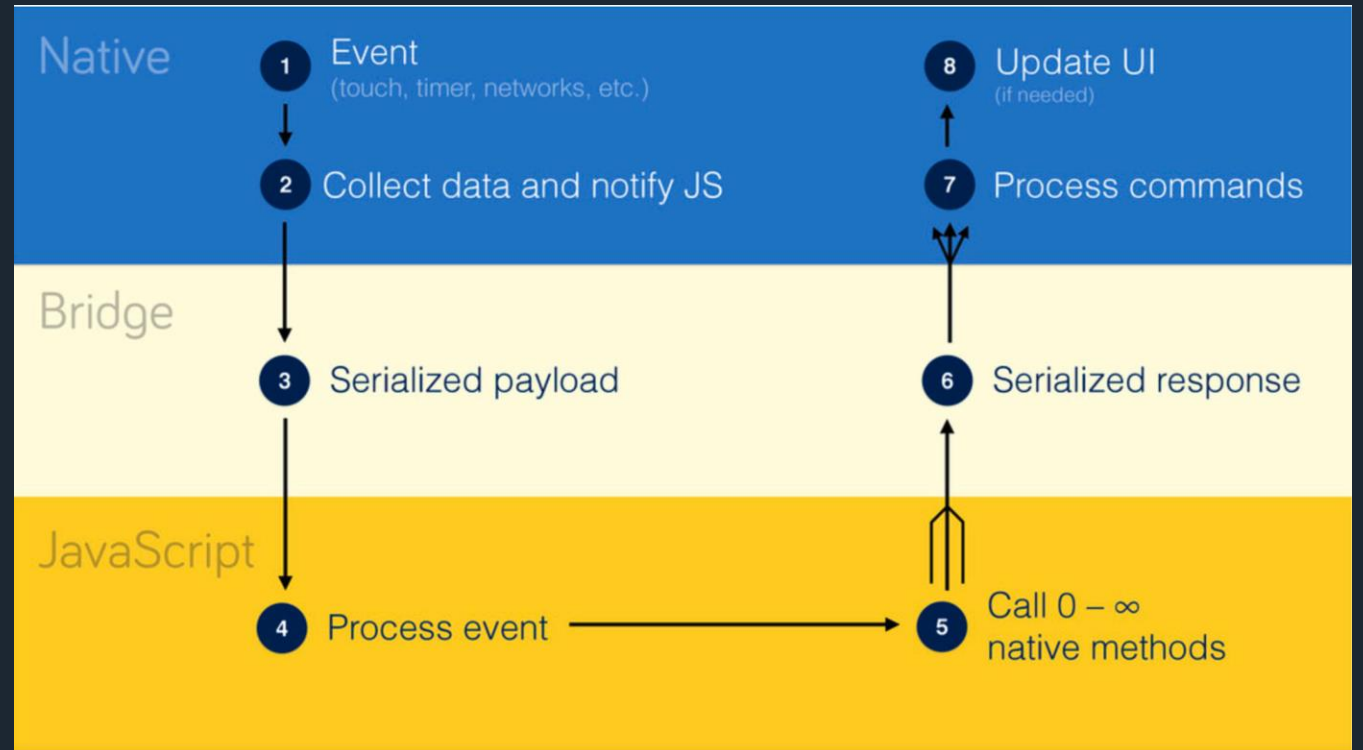
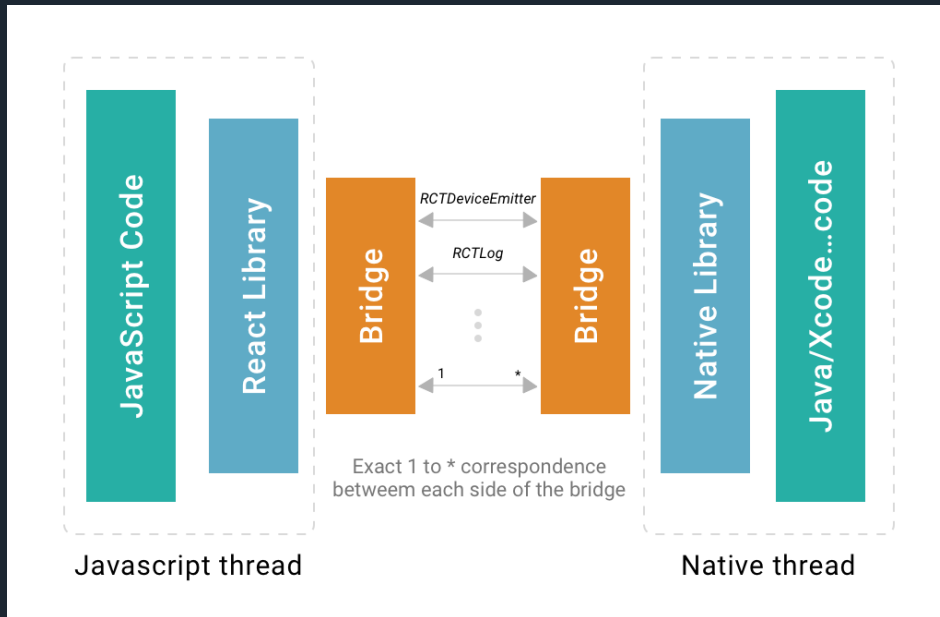
```
}
```



# How it works ?



# How it works ?




# How to run your app ?

- React Native CLI
  - Better control
  - Required configuration from Android Studio or Xcode
- Expo
  - Easy & free access
  - Force to use expo to manage libraries

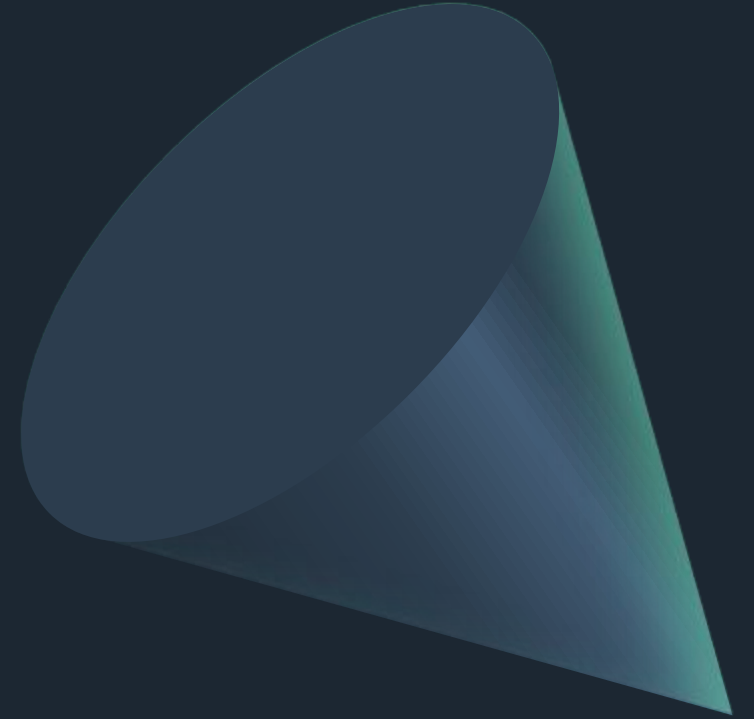


What we  
will learn ?





Basics React Native  
State Management(Redux)  
Navigation  
Native features



# Basics

---

UI Components (View, Button, ...)

---

Stylesheet

---

Flexbox CSS

---

JSX

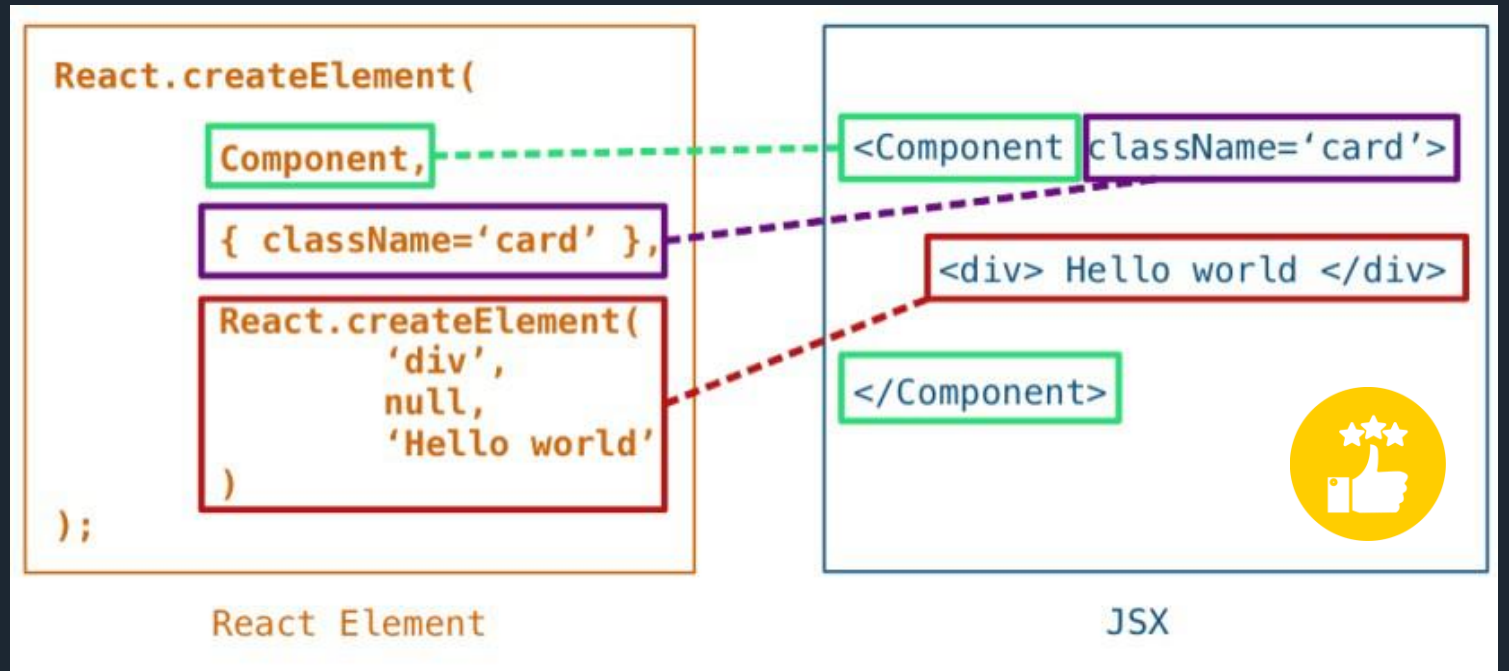
---

Props



# Basics

JSX (Javascript Syntax Extension)



# Basics

## Props

```
type NounoursProps = {  
  1 usage  new *  
  name: string  
}  
  
no usages  new *  
export function NounoursComponent(props: NounoursProps) {  
  return (  
    <View>  
      <Text>Nom : {props.name}</Text>  
    </View>  
  );  
}
```



Destructuring (allowed for screens) :

```
export function NounoursComponent({name}: { name: string }) {  
  return (  
    <View>  
      <Text>Nom : {name}</Text>  
    </View>  
  );  
}
```

# Basics

Components

Component

```
type NounoursProps = {  
  2 usages new *  
  name: string  
}
```

Props

```
no usages new *  
export function NounoursComponent(props: NounoursProps) {  
  
  const nounoursName = props.name;  
  
  return (  
    <View style={nounoursStyle.container}>  
      <Text style={nounoursStyle.title}>Mon nounours préféré</Text>  
      <Text>Nom : {nounoursName}</Text>  
    </View>  
  );  
}
```

```
const nounoursStyle = StyleSheet.create({  
  container: {  
    flex: 1  
  },  
  title: {  
    fontSize: 42,  
    fontWeight: "bold"  
  }  
});
```

Stylesheet