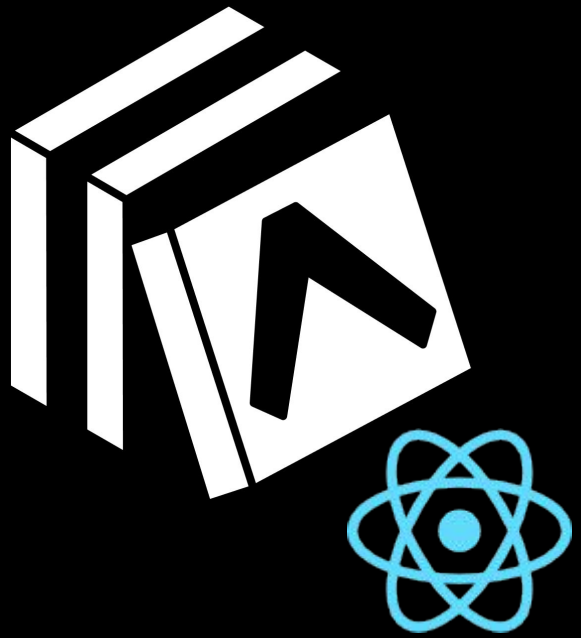
Orange Digital Center

Introduction to React Native & Expo



Orange Digital Center

orangedigitalcenters.com

2024

What is React Native ?

React Native is an open source framework for building Android and iOS applications using React and the app platform’s native capabilities. With React Native, you use JavaScript to access your platform’s APIs as well as to describe the appearance and behavior of your UI using React components.

What is Expo ?

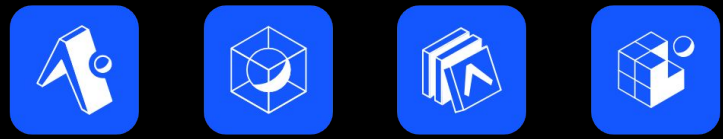
A Programming Language ?

A Framework ?

A Platform ?

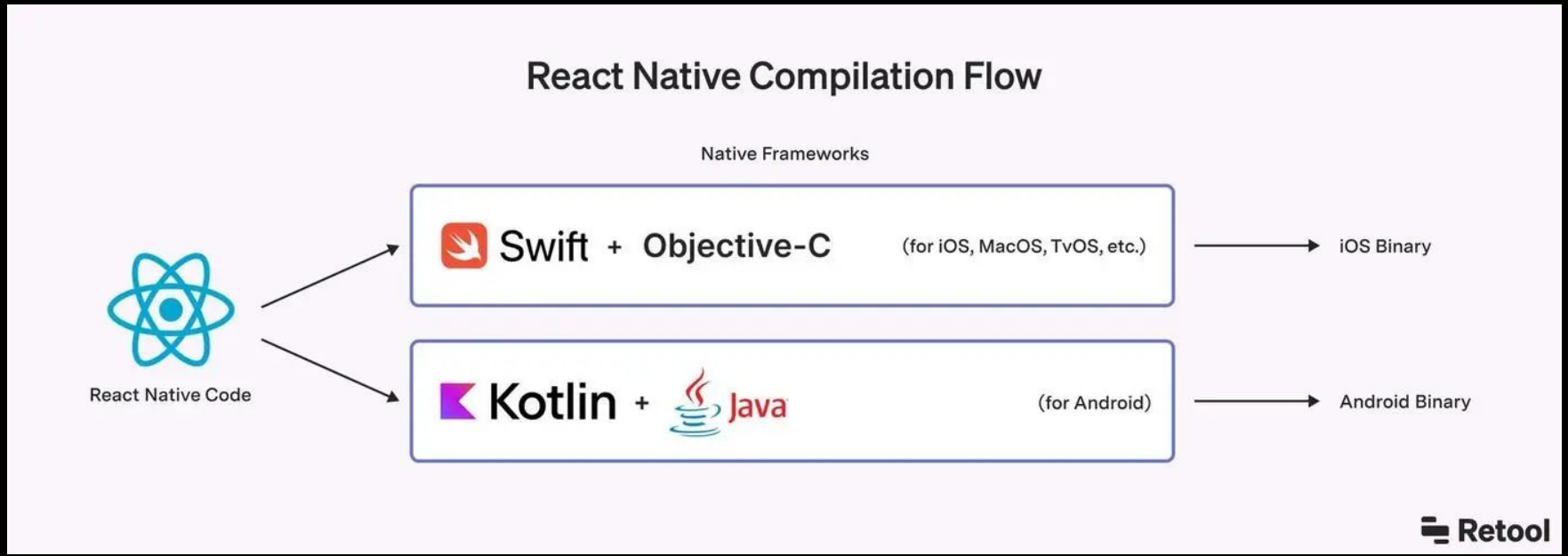
What is Expo ?

Expo is a platform that supports React Native application development. It comes with build and production environments, you can write React Native code using JS (no need for Xcode or Android Studio), and develop for both iOS and Android.



How React Native Works ?

- With React Native, a single JavaScript codebase is natively compiled into platform-specific code for iOS and Android, enabling deployment across multiple device types.
- React Native requires XCode (iOS) and Android Studio to compile the native code into mobile binaries.


React Native Compilation Flow
Native Frameworks
Swift + Objective-C
(for iOS, MacOS, TvOS, etc.)
iOS Binary
React Native Code
Kotlin + Java
(for Android)
Android Binary
Retool

How Expo Works ?

Expo flipped React Native development on its head by replacing how compilation, deployment, and testing worked.

Expo Features

- Expo provides an open-source framework that abstracts away the native compilation process
- Expo Go app allows for quick testing and deployment without needing Xcode or Android Studio

Expo Features

- Expo Dev Clients and Expo Application Services expand deployment options beyond the basic Expo Go experience
- Expo handles the cross-platform logic, build configuration, and app store submission

Apps created using Expo

Bloomberg

and the list goes on…

Advantages of Expo

Benefit 01: Real-Time Development with Expo

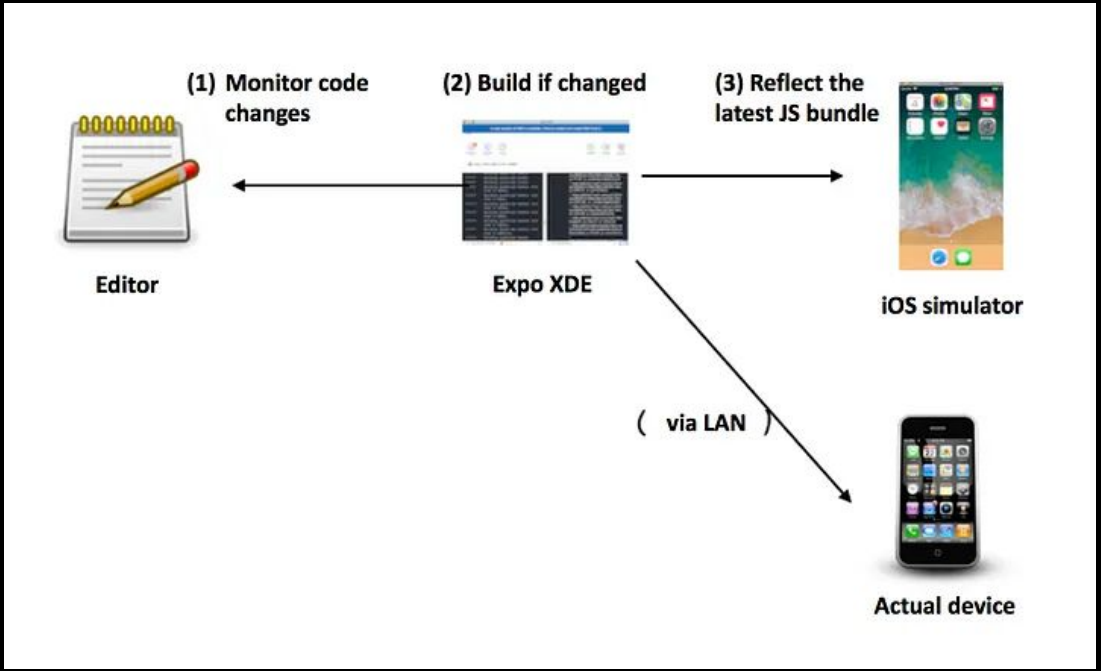
- Instant Code Reflection
- Changes appear immediately in simulator or device Automatic JS compilation
- No rebuilding/restarting needed
- Web-like development experience

Benefit 01: Real-Time Development with Expo

1. **Monitor Code Changes:** Developers write and modify their code and changes made are monitored in real-time.
2. **Build if Changed:** Expo XDE (Expo Development Environment) detects these changes It automatically builds/compiles the new JavaScript bundle.
3. **Reflect the Latest JS Bundle:** The compiled code is then instantly pushed to:

- iOS simulator for testing on a virtual device
- Actual physical device connected via LAN

Benefit 01: Real-Time Development with Expo

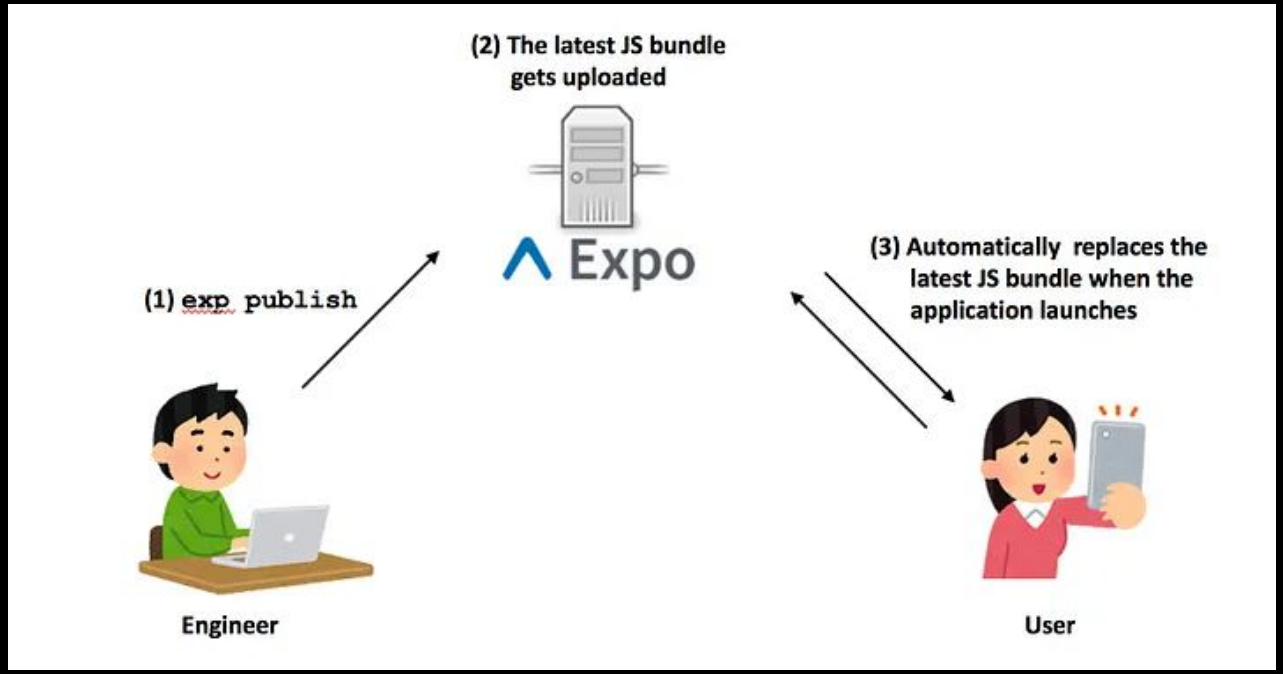


Benefit 02: Over The Air Updates with Expo

- Skip traditional app store review process
- Update both iOS & Android simultaneously

⇒ The key advantage is being able to bypass the lengthy app store submission process by updating just the JavaScript bundle, similar to updating a website.

Benefit 02: Over The Air Updates with Expo



Why React Native ?

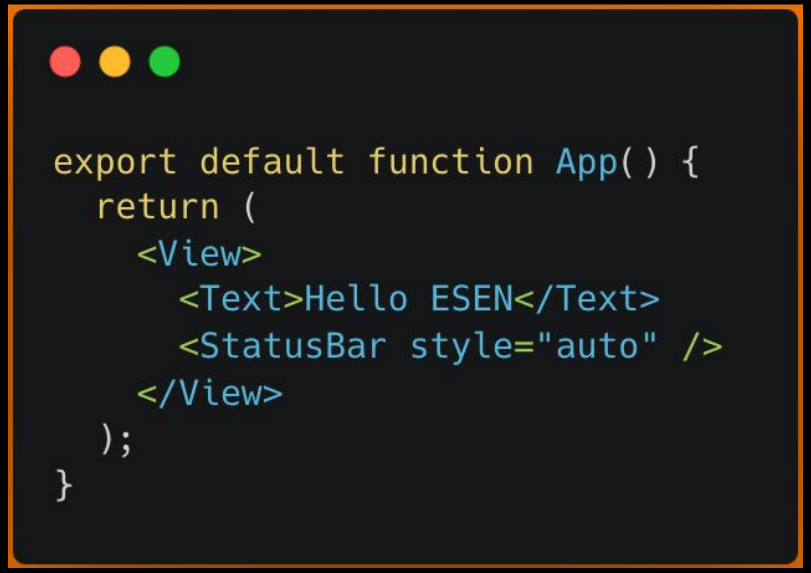
- Cross-Platform Development: Write one codebase for iOS and Android, saving time and resources.
- Native-Like Performance: Uses native components for smooth, high-performance user experiences.
- Hot Reloading: Instantly see code changes, speeding up development and debugging

Why React Native ?

- Component-based: Reusable components make development faster
- JSX Syntax: Allows mixing HTML and JavaScript.
- Lightweight and with an active community

JSX

- JSX, short for JavaScript Syntax eXtension, combines HTML and JavaScript and is based on XML

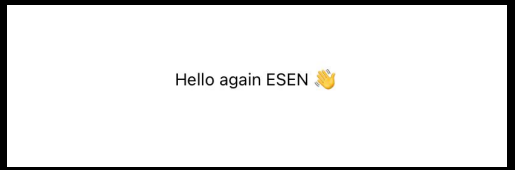


```
export default function App() {
  return (
    <View>
      <Text>Hello ESEN</Text>
      <StatusBar style="auto" />
    </View>
  );
}
```

JSX

```jsx
export default function App() {
  const message = "Hello again ESEN 👋"
  return (
    <View style={styles.container}>
      <Text>{message}</Text>
      <StatusBar style="auto" />
    </View>
  );
}
```

Result :

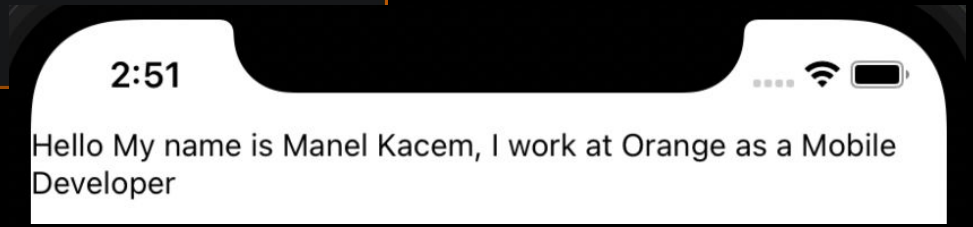


JSX

```jsx
export default function App() {
  const employee = {
    firstName: "Manel",
    lastName: "Kacem",
    work: "Orange",
    job: "Mobile Developer",
  };

  return (
    <SafeAreaView>
      <View>
        <Text>
          Hello My name is {employee.firstName} {employee.lastName}, I work at{" "}
          {employee.work} as a {employee.job}
        </Text>
        <StatusBar style="auto" />
      </View>
    </SafeAreaView>
  );
}
```

Result :



```tsx
export default function App() {
  const listEmployees = [
    {
      firstName: "Manel",
      lastName: "Kacem",
      work: "Orange",
      job: "Mobile Developer",
    },
    {
      firstName: "Flen",
      lastName: "Ben Foulen",
      work: "Orange",
      job: "Mobile Developer",
    },
  ];

  return (
    <SafeAreaView>
      <View>
        {listEmployees.map((employee) => (
          <Text>
            Hello My name is {employee.firstName} {employee.lastName}, I work at{" "}
            {employee.work} as a {employee.job}
          </Text>
        ))}
        <StatusBar style="auto" />
      </View>
    </SafeAreaView>
  );
}
```

Result :

Hello My name is Manel Kacem, I work at Orange as a Mobile Developer
Hello My name is Flen Ben Foulen, I work at Orange as a Mobile Developer

Setting Up the Dev Environment

What you have to install:

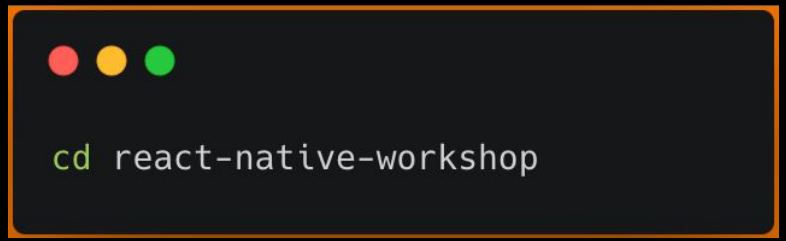
- Install Node.js and npm or yarn
- Install Expo CLI
- A Code Editor such as Visual Studio Code
- Install Expo Go App (for testing)

Steps to Create a New Expo Project

Step 1: Run the following command to start a new React Native JS project.

Steps to Create a New Expo Project

Step 4: Navigate to your project



Steps to Create a New Expo Project

Step 5: Open project with your preferred IDE and start server

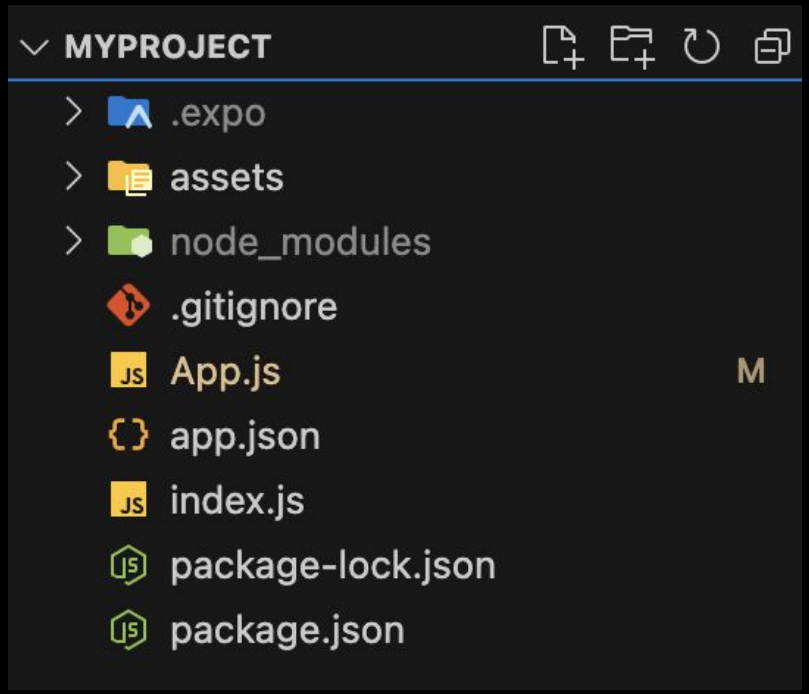
Good job 👏

The results you should be seeing:

Now you can run the app using:

1. Android emulator
2. iOS simulator
3. Expo Go
4. Web

Folder Structure of a React App



- **index.js**: The entry point for the JavaScript bundler (e.g., Metro).
- **App.js**: The main entry point of your application.
- **app.json**: A configuration file for Expo.
- **assets/**:This folder is used to store static assets like images, fonts, and other resources that your app uses.

Thank You

Orange
Digital
Center