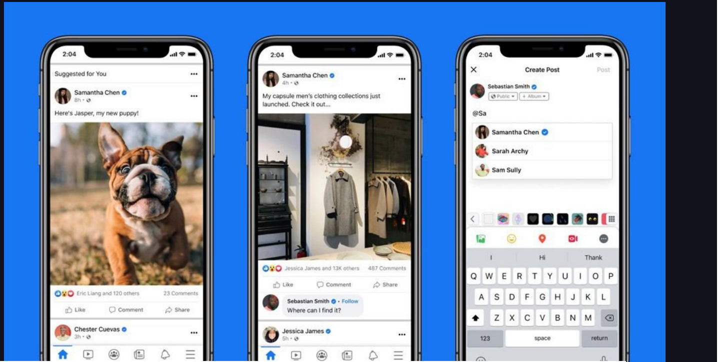
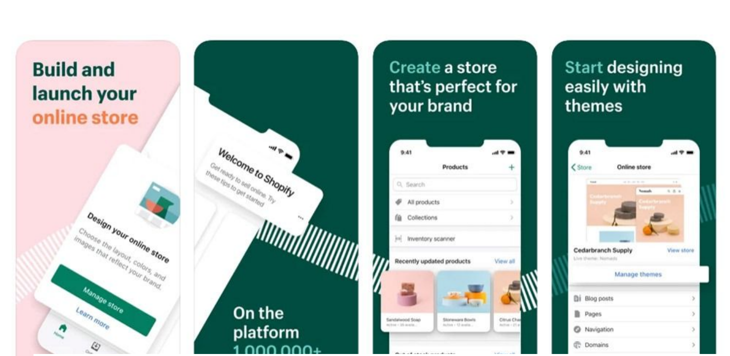
iOS + Android Mobile Apps With React Native (Expo)

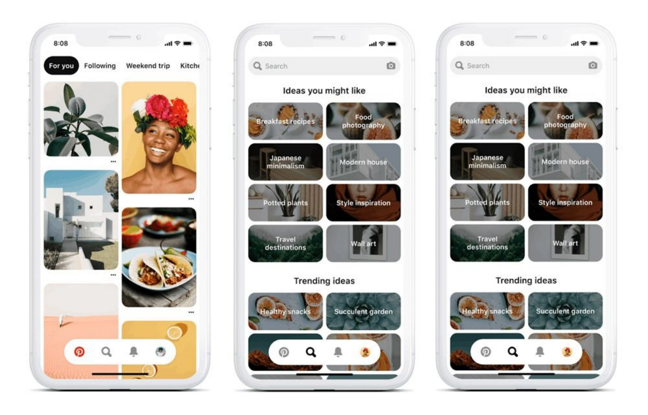
github.com/abubahmed/hoagiehacks-react-native

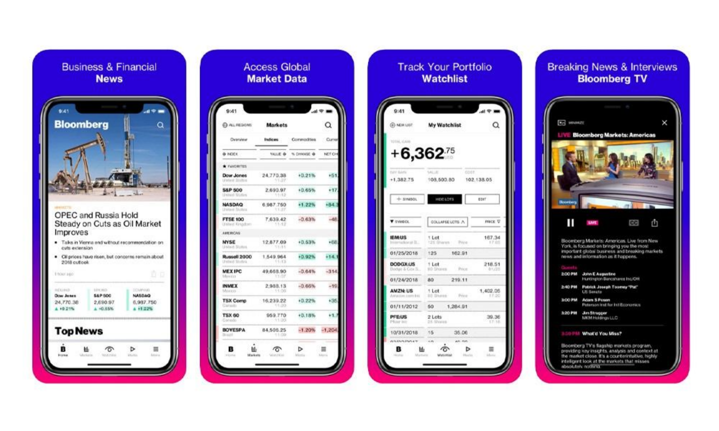
Hoagie Hacks 04/04/2025

Popular RN Apps

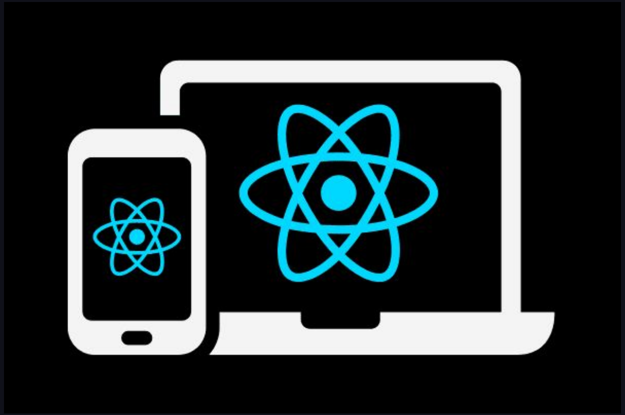



Build and launch your online store
Welcome to Shopify
Design your online store
Manage store
Learn more
On the platform 1,000,000+
Create a store that's perfect for your brand
Products
Start designing easily with themes
Online store
Manage themes




Business & Financial News
Bloomberg
OPEC and Russia Hold Steady on Cuts as Oil Market Improves
Top News
Access Global Market Data
Markets
Track Your Portfolio Watchlist
My Watchlist
+6,362.75
Breaking News & Interviews Bloomberg TV
LIVE Bloomberg Markets: Americas

What is React Native?



- Developed by Meta in 2015
- Used by companies such as Shopify, Walmart, Bloomberg
- Mobile app development framework for making cross-platform apps (iOS + Android) with native elements in JS + React

What is React Native?

JS + React

Uses familiar syntax and coding patterns from web dev (JSX, JS, components)

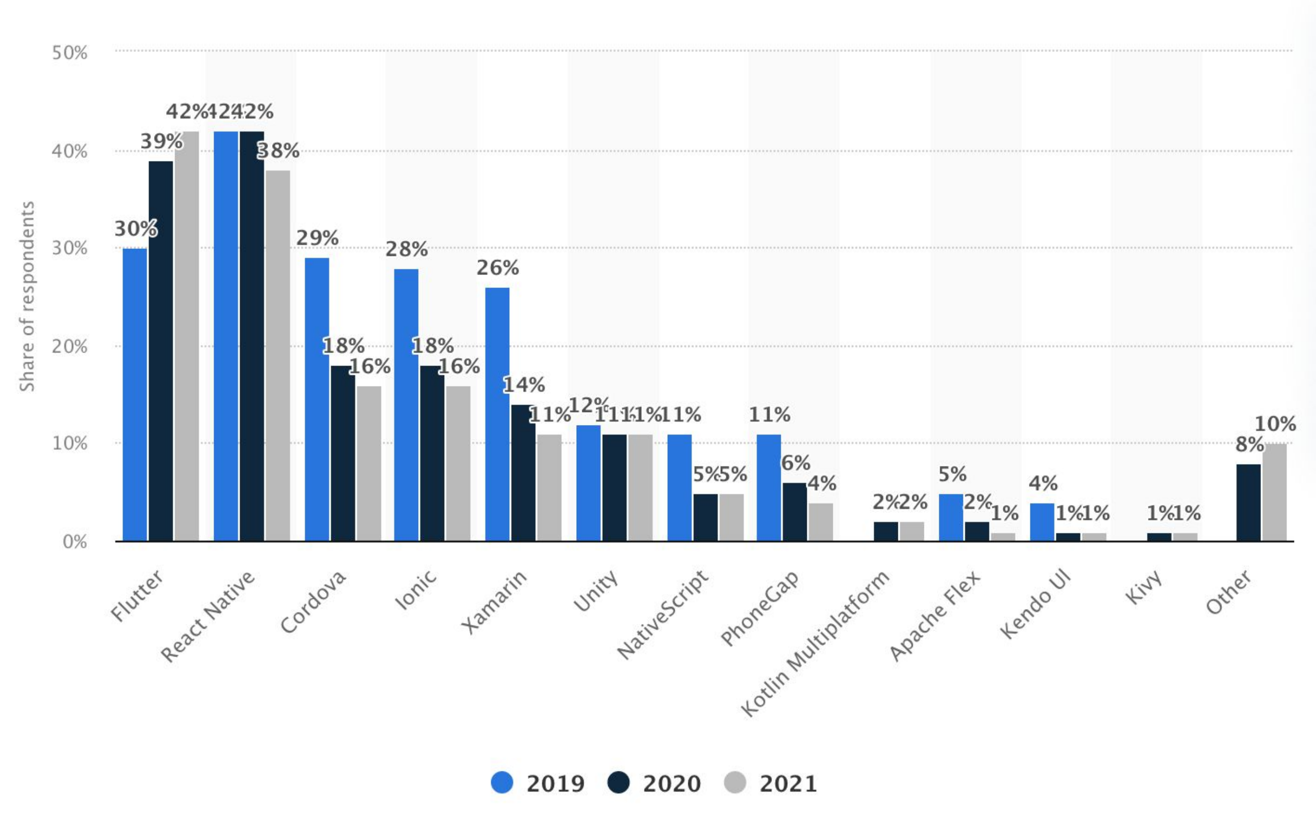
Cross-platform

Build once, deploy to Android and iOS; one single codebase

Native Elements

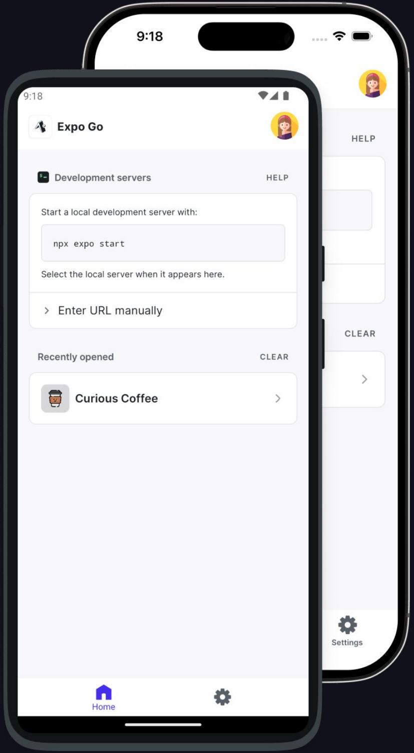
Auto provides native elements - minimal understanding of native dev needed

What is React Native?



What is Expo?

- Toolkit built on top of RN
- Streamlines the setup, development, and deployment process
  - No immediate need for XCode or Android Studio
- Offers more native modules (e.g. camera, notifications)

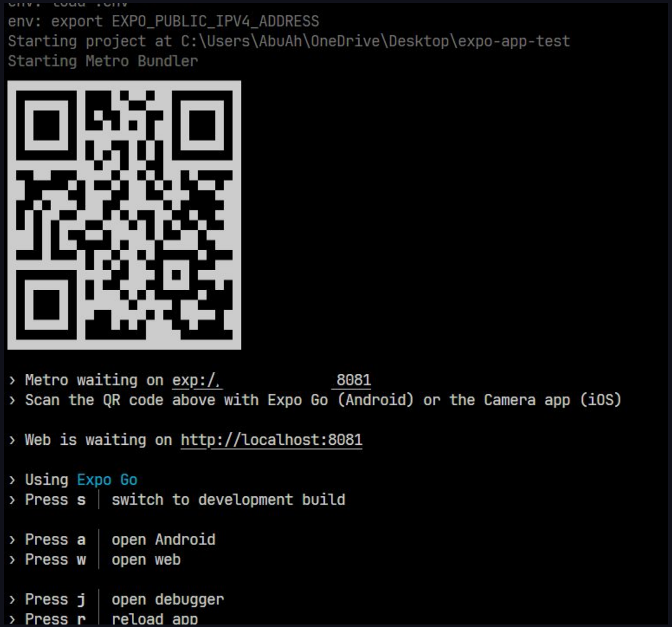


Start a local development server with:

```
npx expo start
```

Select the local server when it appears here.

Creating a RN Expo Project



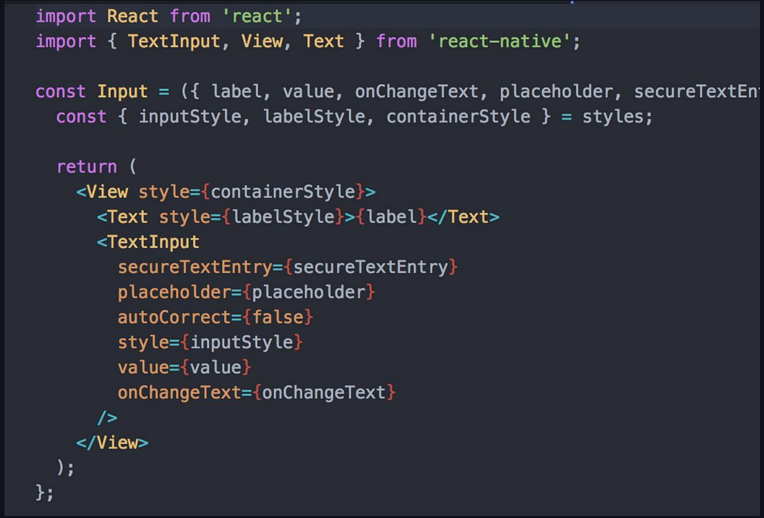
```
env: load .env
env: export EXPO_PUBLIC_IPV4_ADDRESS
Starting project at C:\Users\AbuAh\OneDrive\Desktop\expo-app-test
Starting Metro Bundler

› Metro waiting on exp:/.            8081
› Scan the QR code above with Expo Go (Android) or the Camera app (iOS)

› Web is waiting on http://localhost:8081

› Using Expo Go
› Press s │ switch to development build

› Press a │ open Android
› Press w │ open web

› Press j │ open debugger
› Press r │ reload app
```

https://docs.expo.dev/tutorial/create-your-first-app/

- mkdir your-folder-name
- cd your-folder-name
- npx create-expo-app@latest .
- npm run reset-project
- npx expo start

Writing RN Code

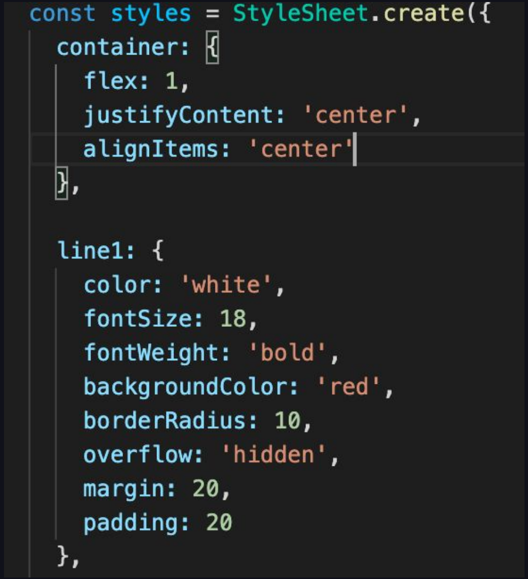
- React Native is built on top of React - same exact syntax with slight differences
  - JSX doesn’t use HTML components (e.g. <div></div>, <p></p>)
  - Instead uses OS native components (<View></View>, <TouchableOpacity></TouchableOpacity>)



```
import React from 'react';
import { TextInput, View, Text } from 'react-native';

const Input = ({ label, value, onChangeText, placeholder, secureTextEn
  const { inputStyle, labelStyle, containerStyle } = styles;

  return (
    <View style={containerStyle}>
      <Text style={labelStyle}>{label}</Text>
      <TextInput
        secureTextEntry={secureTextEntry}
        placeholder={placeholder}
        autoCorrect={false}
        style={inputStyle}
        value={value}
        onChangeText={onChangeText}
      />
    </View>
  );
};
```

https://reactnative.dev/docs/intro-react-native-components

Writing RN Code



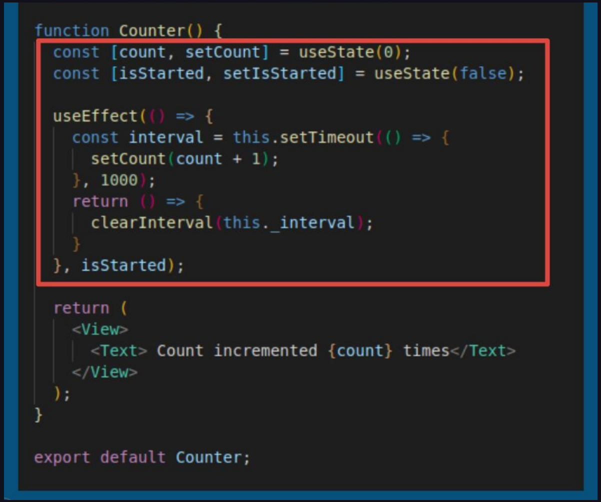
```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center'
  },

  line1: {
    color: 'white',
    fontSize: 18,
    fontWeight: 'bold',
    backgroundColor: 'red',
    borderRadius: 10,
    overflow: 'hidden',
    margin: 20,
    padding: 20
  },
```

https://reactnative.dev/docs/style

- React Native is built on top of React - same exact syntax with slight differences
  - No CSS stylesheets or inline styles - only JS objects
  - No CSS selectors (.class #id)
  - No units (px, rem, vh, vw)
  - No media queries (@media (max-width: 400px))

Writing RN Code

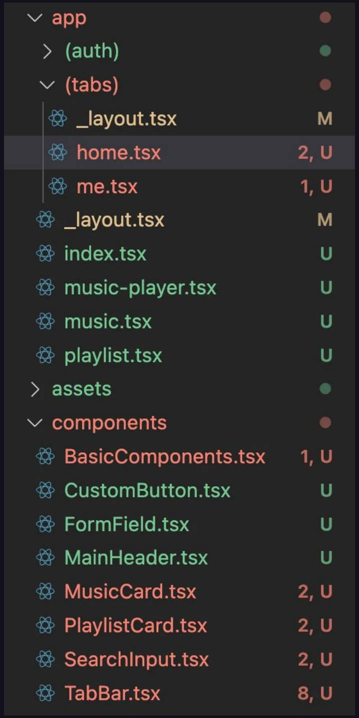
- React Native is built on top of React - same exact syntax with slight differences
    - React hooks (e.g. useState, useEffect, useContext)
    - Component architecture, props, JSX
    - API requests (fetching, Axios)



```
function Counter() {
  const [count, setCount] = useState(0);
  const [isStarted, setIsStarted] = useState(false);

  useEffect(() => {
    const interval = this.setTimeout(() => {
      setCount(count + 1);
    }, 1000);
    return () => {
      clearInterval(this._interval);
    }
  }, isStarted);

  return (
    <View>
      <Text> Count incremented {count} times</Text>
    </View>
  );
}

export default Counter;
```

https://react.dev/reference/react/hooks

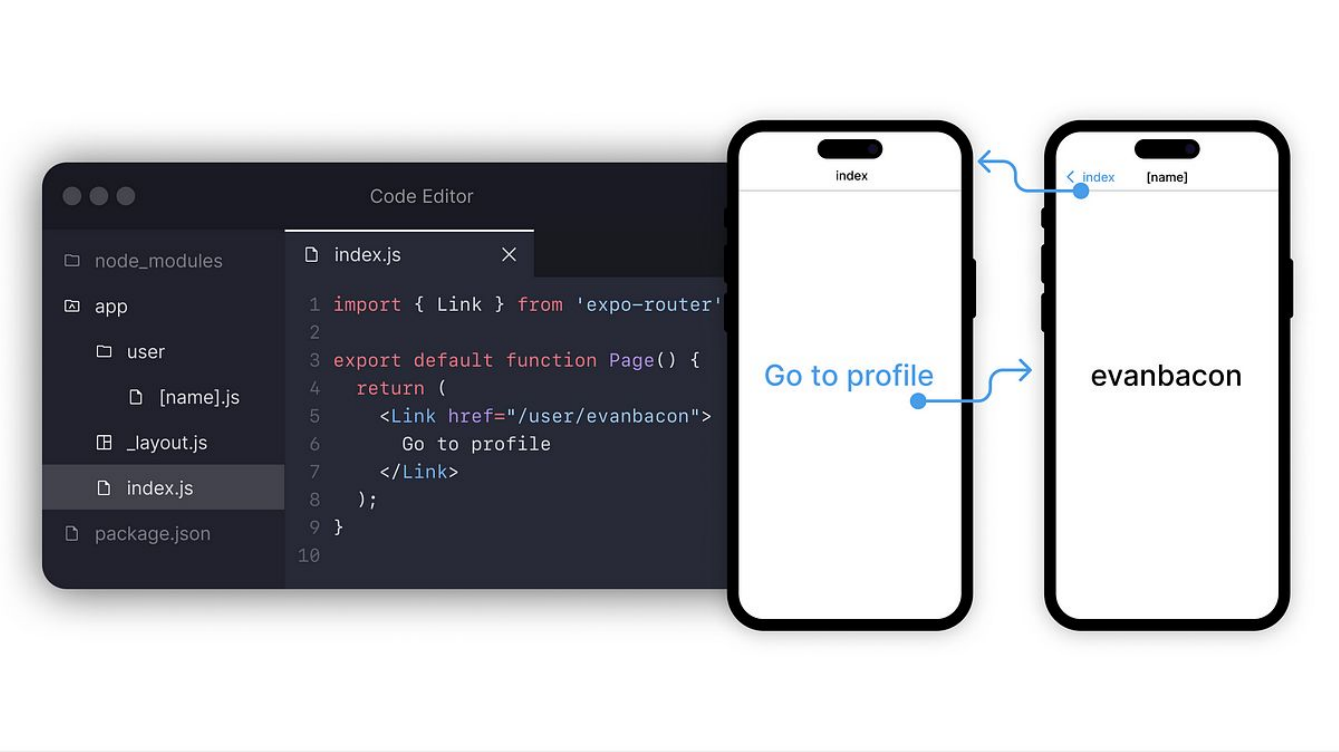
Expo Router



```
⌄ app                          ●
  › (auth)                     ●
  ⌄ (tabs)                     ●
      _layout.tsx              M
      home.tsx              2, U
      me.tsx                1, U
    _layout.tsx                M
    index.tsx                  U
    music-player.tsx           U
    music.tsx                  U
    playlist.tsx               U
› assets                       ●
⌄ components                   ●
    BasicComponents.tsx     1, U
    CustomButton.tsx           U
    FormField.tsx              U
    MainHeader.tsx             U
    MusicCard.tsx           2, U
    PlaylistCard.tsx        2, U
    SearchInput.tsx         2, U
    TabBar.tsx              8, U
```

- Screen routing infrastructure - more akin to Next.js than plain React.js
- Folder-structure based routing system: your folder structure defines the routes in your app
  - e.g. details.tsx -> /details -> details screen
- Also provides support for stacks, tabs, drawers

Expo Router



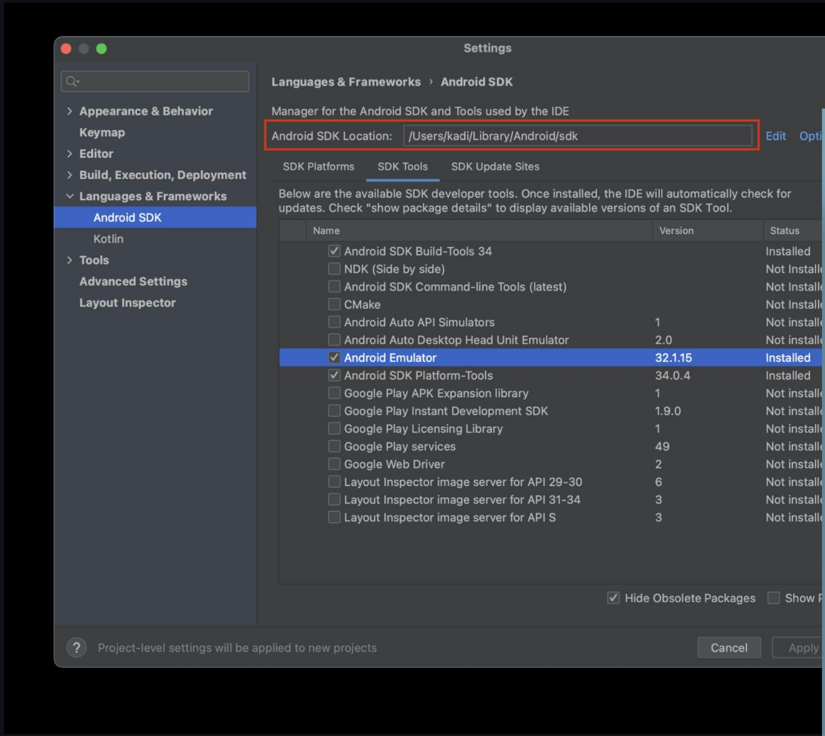
```
import { Link } from 'expo-router'

export default function Page() {
  return (
    <Link href="/user/evanbacon">
      Go to profile
    </Link>
  );
}

```

https://docs.expo.dev/router/introduction/

Powerful Expo/RN APIs

- expo-camera
- expo-location
- expo-notifications
- expo-media-library
- expo-secure-store
- expo-haptics
- react-native-maps
- expo-sensors
- expo-print
- expo-file-system
- @react-native-community/datetimepicker

Android Studio/Xcode



Languages & Frameworks › Android SDK

Manager for the Android SDK and Tools used by the IDE

Android SDK Location: /Users/kadi/Library/Android/sdk Edit Opti

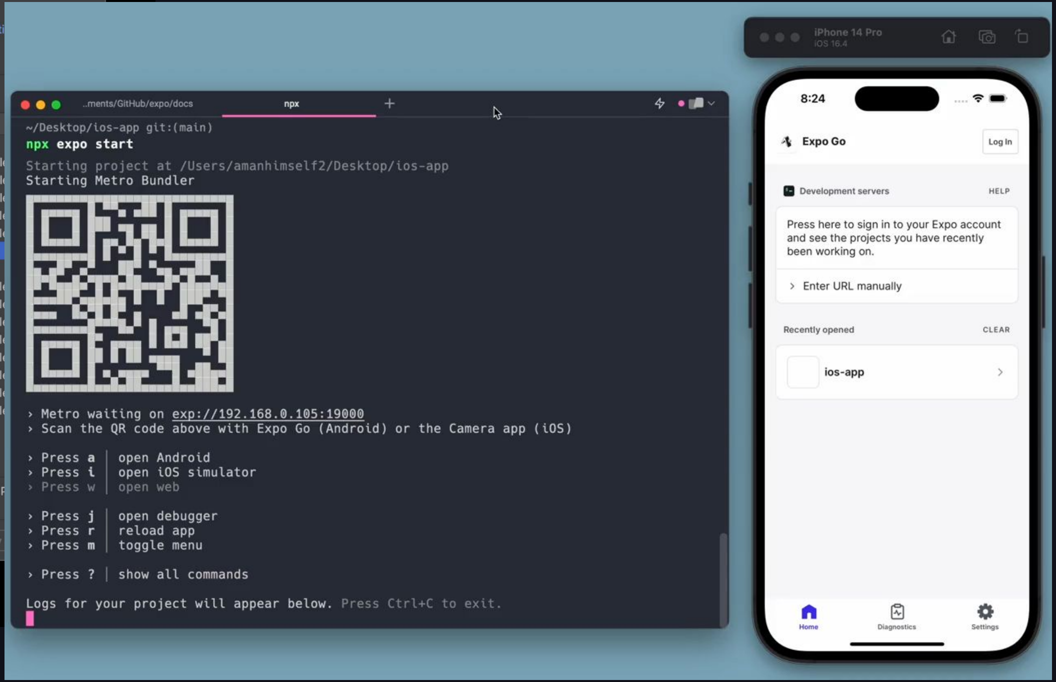
SDK Platforms | SDK Tools | SDK Update Sites

Below are the available SDK developer tools. Once installed, the IDE will automatically check for updates. Check "show package details" to display available versions of an SDK Tool.

Name	Version	Status
☑ Android SDK Build-Tools 34		Installed
☐ NDK (Side by side)		Not Installe
☐ Android SDK Command-line Tools (latest)		Not Installe
☐ CMake		Not Installe
☐ Android Auto API Simulators	1	Not installe
☐ Android Auto Desktop Head Unit Emulator	2.0	Not installe
☑ Android Emulator	32.1.15	Installed
☑ Android SDK Platform-Tools	34.0.4	Installed
☐ Google Play APK Expansion library	1	Not installe
☐ Google Play Instant Development SDK	1.9.0	Not installe
☐ Google Play Licensing Library	1	Not installe
☐ Google Play services	49	Not installe
☐ Google Web Driver	2	Not installe
☐ Layout Inspector image server for API 29-30	6	Not installe
☐ Layout Inspector image server for API 31-34	3	Not installe
☐ Layout Inspector image server for API S	3	Not installe

☑ Hide Obsolete Packages ☐ Show P

? Project-level settings will be applied to new projects



```
~/Desktop/ios-app git:(main)
npx expo start
Starting project at /Users/amanhimself2/Desktop/ios-app
Starting Metro Bundler

› Metro waiting on exp://192.168.0.105:19000
› Scan the QR code above with Expo Go (Android) or the Camera app (iOS)

› Press a │ open Android
› Press i │ open iOS simulator
› Press w │ open web

› Press j │ open debugger
› Press r │ reload app
› Press m │ toggle menu

› Press ? │ show all commands

Logs for your project will appear below. Press Ctrl+C to exit.
```

Press here to sign in to your Expo account and see the projects you have recently been working on.