

Expo

EXPO



É um conjunto de ferramentas e serviços construídos em torno de plataformas nativas e React Native que ajudam você a desenvolver, construir, implantar e iterar rapidamente em aplicativos iOS, Android e web a partir da mesma base de código JavaScript/TypeScript.

EXPO

Expo

Search



Docs

Tools

EAS

Pricing

Careers

Sign Up

Log In

Make any app. Run it everywhere.

Build one project that runs natively
on all your users' devices

Get Started →

Join 400K+ developers including

dailypay.

facepunch

few

flexport.

Front

goody

EXPO



Terminal

 Copy

```
# Create a project named my-app  
- npx create-expo-app my-app  
  
# Navigate to the project directory  
- cd my-app
```

Terminal

 Copy

```
- npx expo start
```

EXPO

✓ Downloaded and extracted project files.

> npm install

npm WARN deprecated @npmcli/move-file@1.1.2: This functionality has been moved to @npmcli/fs

npm WARN deprecated source-map-url@0.4.1: See <https://github.com/lydell/source-map-url#deprecated>

npm WARN deprecated urix@0.1.0: Please see <https://github.com/lydell/urix#deprecated>

npm WARN deprecated resolve-url@0.2.1: <https://github.com/lydell/resolve-url#deprecated>

npm WARN deprecated source-map-resolve@0.5.3: See <https://github.com/lydell/source-map-resolve#deprecated>

npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See <https://v8.dev/blog/math-random> for details.

npm WARN deprecated uglify-es@3.3.9: support for ECMAScript is superseded by `uglify-js` as of v3.13.0

added 1239 packages, and audited 1240 packages in 12s

61 packages are looking for funding

run `npm fund` for details

5 **high** severity vulnerabilities

To address all issues (including breaking changes), run:

npm audit fix --force

Run `npm audit` for details.

✓ **Your project is ready!**

To run your project, navigate to the directory and run one of the following npm commands.

- **cd my-app**
- **npm run android**
- **npm run ios**
- **npm run web**

```
anderson@MacBook-Pro-de-Anderson my-app % npm run android
```

```
> my-app@1.0.0 android  
> expo start --android
```

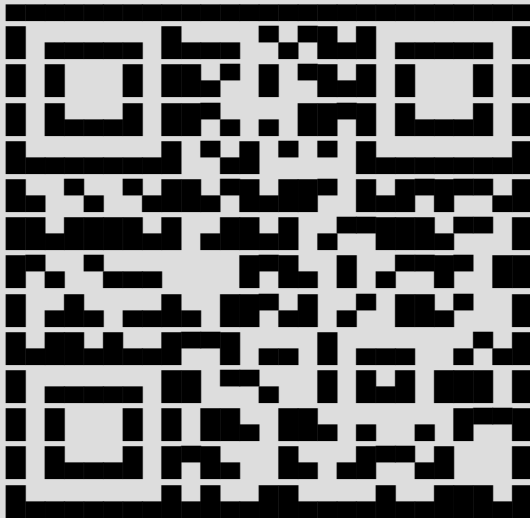
```
Starting project at /Users/anderson/my-app
```

```
Starting Metro Bundler
```

```
> Opening exp://192.168.0.22:19000 on Pixel
```

```
✓ Expo Go on Pixel is outdated, would you like to upgrade? ... yes
```

```
Uninstalling Expo Go from android device Pixel.
```



```
> Metro waiting on exp://192.168.0.22:19000
```

```
> 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.
```

```
Android Bundling complete 15261ms
```



```
anderson@MacBook-Pro-de-Anderson my-app % npm run android
```

```
> my-app@1.0.0 android  
> expo start --android
```

EXPO

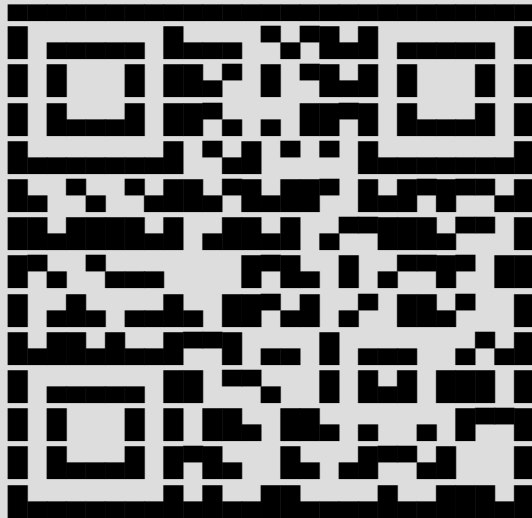
```
Starting project at /Users/anderson/my-app
```

```
Starting Metro Bundler
```

```
> Opening exp://192.168.0.22:19000 on Pixel
```

```
✓ Expo Go on Pixel is outdated, would you like to upgrade? ... yes
```

```
Uninstalling Expo Go from android device Pixel.
```



```
> Opening on iOS...  
> Opening exp://192.168.0.22:19000 on iPhone 14 Pro Max  
Downloading the Expo Go app [=====] 100% 0.0s
```

```
> Metro waiting on exp://192.168.0.22:19000
```

```
> 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.
```

```
Android Bundling complete 15261ms
```

COMPONENTES FUNCIONAIS



Os componentes em React que declarados em formato de função Javascript, são chamados de Componentes Funcionais, estes assumem o seguinte formato:

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

const App = () => {
  return (
    <View>
      <Text>Hello Samurai</Text>
    </View>
  );
};

export default App;
```

Tendo como notáveis características, ser uma função Javascript que retorna um componente visual para ser exibido na tela.

USESTATE



A ideia é semelhante a um atributo de classe e seu método Set.

```
const [variavel, setVariavel] = useState(valorInicial)
```

Onde:

variável = nome da variável.

setVariavel = função para alterar o valor da variável.

valorInicial = valor inicial da variável.

USESTATE



```
import React, { useState } from "react";
import { View, Text, Button } from "react-native";

const Counter = () => {
  const [counter, setCounter] = useState(0);

  return (
    <View>
      <Text>{counter}</Text>
      <Button title="clique aqui" onPress={() => setCounter(1)} />
    </View>
  );
};

const App = () => {
  return (
    <View>
      <Counter />
    </View>
  );
};

export default App;
```

USESTATE



Importação das
Bibliotecas

```
import React, { useState } from "react";  
import { View, Text, Button } from "react-native";
```

```
const Counter = () => {  
  const [counter, setCounter] = useState(0);  
  
  return (  
    <View>  
      <Text>{counter}</Text>  
      <Button title="clique aqui" onPress={() => setCounter(1)} />  
    </View>  
  );  
};
```

```
const App = () => {  
  return (  
    <View>  
      <Counter />  
    </View>  
  );  
};
```

```
export default App;
```

Fonte: <https://devsamurai.com.br/react-native-basics-props-states/>

USESTATE



```
import React, { useState } from "react";  
import { View, Text, Button } from "react-native";
```

```
const Counter = () => {  
  const [counter, setCounter] = useState(0);  
  
  return (  
    <View>  
      <Text>{counter}</Text>  
      <Button title="clique aqui" onPress={() => setCounter(1)} />  
    </View>  
  );  
};
```

Componente Counter

```
const App = () => {  
  return (  
    <View>  
      <Counter />  
    </View>  
  );  
};
```

```
export default App;
```

Fonte: <https://devsamurai.com.br/react-native-basics-props-states/>

USESTATE



```
import React, { useState } from "react";  
import { View, Text, Button } from "react-native";
```

```
const Counter = () => {  
  const [counter, setCounter] = useState(0);
```

setCounter

```
  return (  
    <View>  
      <Text>{counter}</Text>  
      <Button title="clique aqui" onPress={() => setCounter(1)} />  
    </View>  
  );  
};
```

```
const App = () => {  
  return (  
    <View>  
      <Counter />  
    </View>  
  );  
};
```

```
export default App;
```

USESTATE



```
import React, { useState } from "react";  
import { View, Text, Button } from "react-native";
```

```
const Counter = () => {  
  const [counter, setCounter] = useState(0);
```

```
  return (  
    <View>  
      <Text>{counter}</Text>  
      <Button title="clique aqui" onPress={() => setCounter(1)} />  
    </View>  
  );  
};
```

O que é o componente Counter

```
const App = () => {  
  return (  
    <View>  
      <Counter />  
    </View>  
  );  
};
```

```
export default App;
```

USESTATE



```
import React, { useState } from "react";
import { View, Text, Button } from "react-native";
```

```
const Counter = () => {
  const [counter, setCounter] = useState(0);
```

```
  return (
```

```
    <View>
```

```
      <Text>{counter}</Text>
```

```
      <Button title="clique aqui" onPress={() => setCounter(1)} />
```

```
    </View>
```

```
  );
```

```
};
```

```
const App = () => {
```

```
  return (
```

```
    <View>
```

```
      <Counter />
```

```
    </View>
```

```
  );
```

```
};
```

```
export default App;
```

Componente Text e a variável counter

USESTATE



```
import React, { useState } from "react";
import { View, Text, Button } from "react-native";
```

```
const Counter = () => {
  const [counter, setCounter] = useState(0);
```

```
  return (
    <View>
```

```
      <Text>{counter}</Text>
```

```
      <Button title="clique aqui" onPress={() => setCounter(1)} />
```

```
    </View>
```

```
  );
```

```
};
```

```
const App = () => {
```

```
  return (
```

```
    <View>
```

```
      <Counter />
```

```
    </View>
```

```
  );
```

```
};
```

```
export default App;
```

Componente Button e o método onPress

USESTATE



```
import React, { useState } from "react";
import { View, Text, Button } from "react-native";

const Counter = () => {
  const [counter, setCounter] = useState(0);

  return (
    <View>
      <Text>{counter}</Text>
      <Button title="clique aqui" onPress={() => setCounter(1)} />
    </View>
  );
};
```

```
const App = () => {
  return (
    <View>
      <Counter />
    </View>
  );
};
```

```
export default App;
```

Interface da minha aplicação

USESTATE



```
import React, { useState } from "react";
import { View, Text, Button } from "react-native";

const Counter = () => {
  const [counter, setCounter] = useState(0);

  return (
    <View>
      <Text>{counter}</Text>
      <Button title="clique aqui" onPress={() => setCounter(1)} />
    </View>
  );
};

const App = () => {
  return (
    <View>
      <Counter />
    </View>
  );
};
```

```
export default App;
```

Aquí estou "exportando" o kernel da minha aplicação

USESTATE



```
import React, { useState } from "react";  
import { View, Text, Button } from "react-native";
```

```
const Counter = () => {  
  const [counter, setCounter] = useState(0);
```

```
  return (  
    <View>  
      <Text>{counter}</Text>  
      <Button title="clique aqui" onPress={() => setCounter(counter + 1)} />  
    </View>  
  );  
};
```

```
const App = () => {  
  return (  
    <View>  
      <Counter />  
    </View>  
  );  
};
```

```
export default App;
```

<Button title="clique aqui" onPress={() => setCounter(1)} />

HOOKS PERSONALIZADOS



```
const useCounter = () => {  
  const [value, setValue] = useState(0);  
  
  return [value, setValue];  
};
```

```

import React, { useState } from "react";
import { View, Text, Button, TextInput, StyleSheet } from "react-native";

const useFatorial = () => {
  const [value, setValue] = useState(0);

  const setFatorial = (newValue) => {
    fatorial=1;
    n=parseInt(newValue);
    if(n<2) setFatorial(fatorial);
    for(i=2; i<=n; i++)
      fatorial = fatorial*i;
    setValue(fatorial);
  };

  return [value, setFatorial];
};

const Fatorial = () => {
  const [fatorial, setFatorial] = useFatorial(0);
  const [n, setN] = useState(0);
  return (
    <View>
      <TextInput
        style={styles.title}
        placeholder="Informe n"
        onChangeText={n => setN(n)}
        keyboardType="numeric"
      />
      <Text>{fatorial}</Text>
      <Button title="clique aqui" onPress={() => setFatorial(n)} />
    </View>
  );
};

const App = () => {
  return (
    <View>
      <Fatorial />
    </View>
  );
};

const styles = StyleSheet.create({
  container: {
    flex: 1,
    padding: 24,
    backgroundColor: '#eaeaea',
  },
  title: {
    marginTop: 16,
    paddingVertical: 80,
  },
});

export default App;

```

```
import React, { useState } from "react";  
import { View, Text, Button, TextInput, StyleSheet } from "react-native";
```

HOOKS PERSONALIZADOS



```
import React, { useState } from "react";
import { View, Text, Button, TextInput, StyleSheet } from "react-native";

const useFatorial = () => {
  const [value, setValue] = useState(0);

  const setFatorial = (newValue) => {
    fatorial=1;
    n=parseInt(newValue);
    if(n<2) setFatorial(fatorial);
    for(i=2; i<=n; i++)
      fatorial = fatorial*i;
    setValue(fatorial);
  };

  return [value, setFatorial];
};
```

```
import React, { useState } from "react";
import { View, Text, Button, TextInput, StyleSheet } from "react-native";

const useFatorial = () => {
  const [value, setValue] = useState(0);

  const setFatorial = (newValue) => {
    fatorial=1;
    n=parseInt(newValue);
    if(n<2) setFatorial(fatorial);
    for(i=2; i<=n; i++)
      fatorial = fatorial*i;
    setValue(fatorial);
  };

  return [value, setFatorial];
};

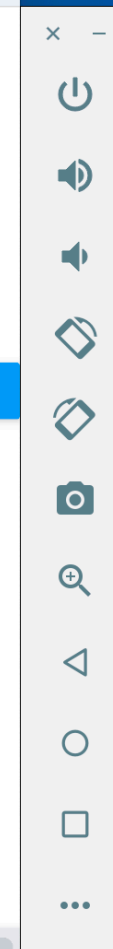
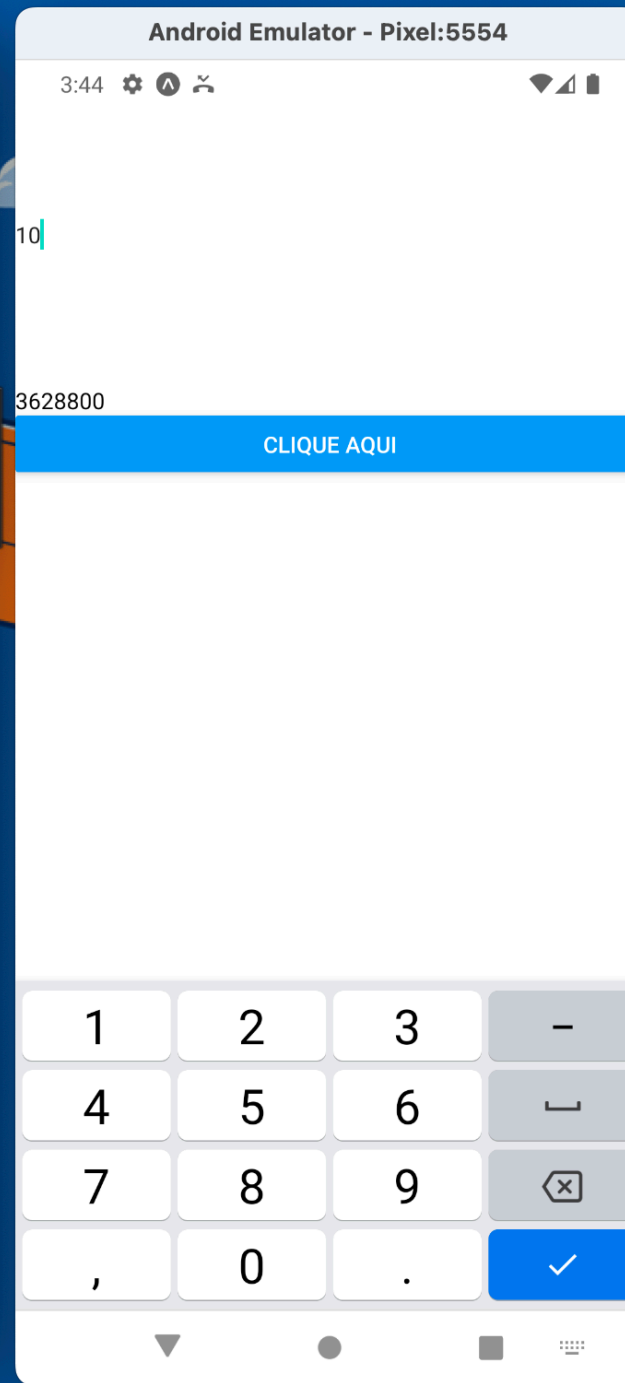
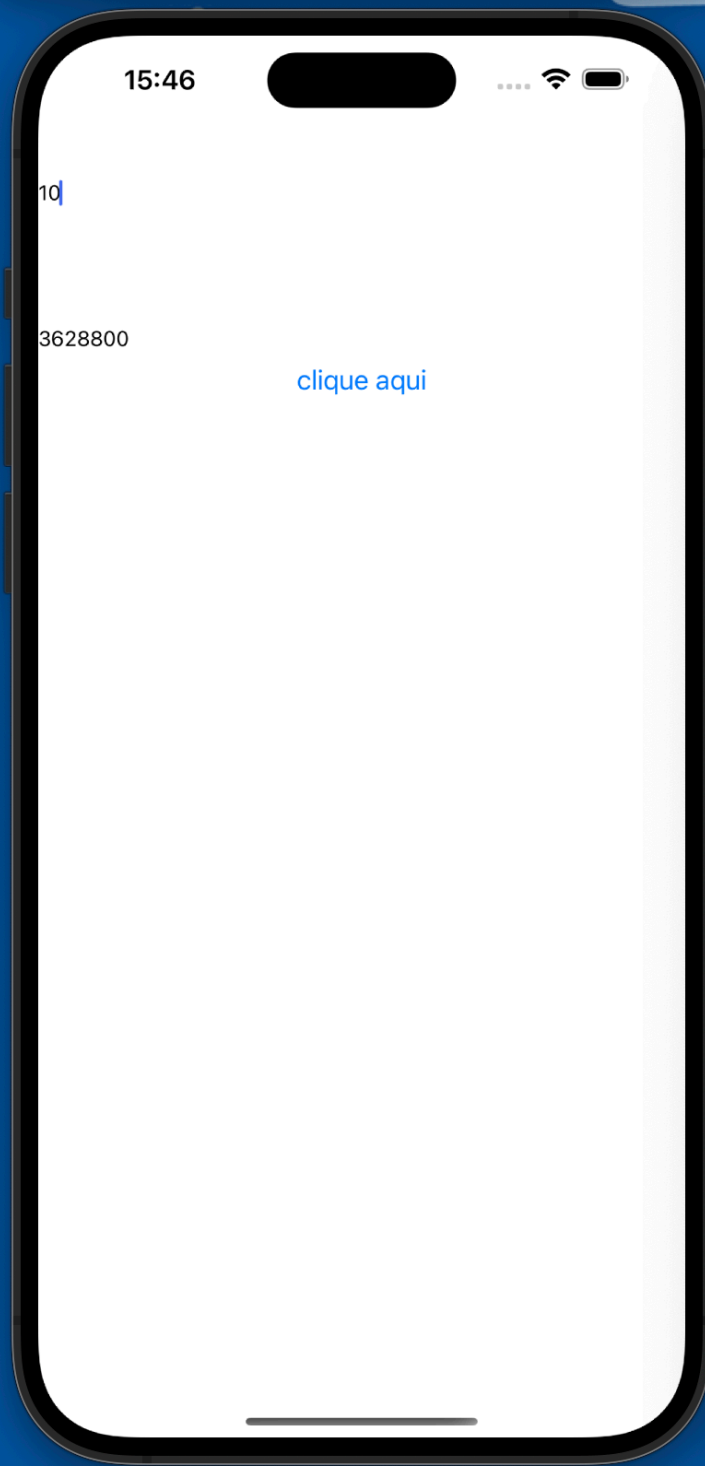
const Fatorial = () => {
  const [fatorial, setFatorial] = useFatorial(0);
  const [n, setN] = useState(0);
  return (
    <View>
      <TextInput
        style={styles.title}
        placeholder="Informe n"
        onChangeText={n => setN(n)}
        keyboardType="numeric"
      />
      <Text>{fatorial}</Text>
      <Button title="clique aqui" onPress={() => setFatorial(n)} />
    </View>
  );
};
```



```
const Fatorial = () => {
  const [fatorial, setFatorial] = useFatorial(0);
  const [n, setN] = useState(0);
  return (
    <View>
      <TextInput
        style={styles.title}
        placeholder="Informe n"
        onChangeText={n => setN(n)}
        keyboardType="numeric"
      />
      <Text>{fatorial}</Text>
      <Button title="clique aqui" onPress={() => setFatorial(n)} />
    </View>
  );
};

const App = () => {
  return (
    <View>
      <Fatorial />
    </View>
  );
};
```

```
const App = () => {  
  return (  
    <View>  
      <Fatorial />  
    </View>  
  );  
};  
  
const styles = StyleSheet.create({  
  container: {  
    flex: 1,  
    padding: 24,  
    backgroundColor: '#eaeaea',  
  },  
  title: {  
    marginTop: 16,  
    paddingVertical: 80,  
  },  
});  
  
export default App;
```



BIBLIOTECAS DE COMPONENTES



- React Native Paper
- React Native Elements
- NativeBase
- React Native UI Kitten
- RNUI: React Native UI Library
- Teaset
- Shoutem UI
- Lottie for React Native
- React Native Maps
- React Native Gifted Chat

Fonte: <https://blog.logrocket.com/react-native-component-libraries/>

BIBLIOTECAS DE COMPONENTES



React Native Paper

Docs

Components

Showcase

v5.x ▾



Q Search



K



Cross-platform Material Design for React Native

Paper is a collection of customizable and production-ready components for React Native, following Google's Material Design guidelines.

Get started

Try on Snack

Or check the demo app on [iOS](#) or [Android](#).

⌂ Loading

📷 Icon

📷 Press me

+

+

+

MD



Display Large

Display Medium

Lorem ipsum dolor, sit amet consectetur adipisicing elit. Est tenetur neque laudantium, repellendus at excepturi quasi qui culpa. Incidunt nesciunt unde perspiciatis atque rerum blanditiis sint ratione, sequi totam temporibus.

PAPER

Instalação: `npm install react-native-paper`

Fonte: <https://callstack.github.io/react-native-paper/4.0/getting-started.html>



```
anderson@MacBook-Pro-de-Anderson my-app % npm install react-native-paper
```

```
added 16 packages, and audited 1256 packages in 2s
```

```
61 packages are looking for funding
  run `npm fund` for details
```

```
5 high severity vulnerabilities
```

```
To address all issues (including breaking changes), run:
  npm audit fix --force
```

```
Run `npm audit` for details.
```

```
anderson@MacBook-Pro-de-Anderson my-app % npm run android
```

```
> my-app@1.0.0 android
> expo start --android
```

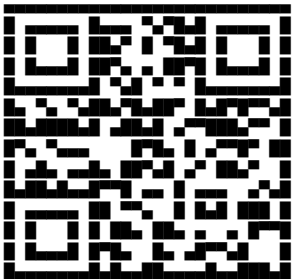
```
Starting project at /Users/anderson/my-app
```

```
Starting Metro Bundler
```

```
> Opening exp://192.168.0.22:19000 on Pixel
```

```
✓ Expo Go on Pixel is outdated, would you like to upgrade? ... yes
```

```
Uninstalling Expo Go from android device Pixel.
```



```
> Metro waiting on exp://192.168.0.22:19000
```

```
> 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.
```

```
anderson@MacBook-Pro-de-Anderson my-app % npm install react-native-paper
added 16 packages, and audited 1256 packages in 2s

61 packages are looking for funding
  run `npm fund` for details

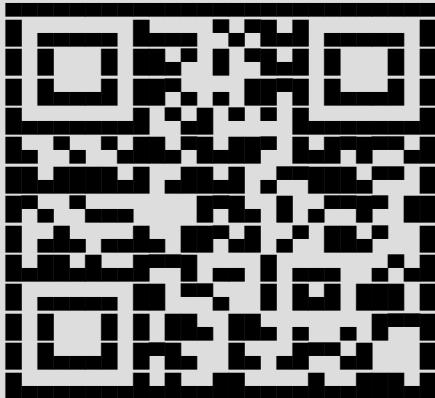
5 high severity vulnerabilities

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
anderson@MacBook-Pro-de-Anderson my-app % npm run android

> my-app@1.0.0 android
> expo start --android

Starting project at /Users/anderson/my-app
Starting Metro Bundler
> Opening exp://192.168.0.22:19000 on Pixel
✓ Expo Go on Pixel is outdated, would you like to upgrade? ... yes
Uninstalling Expo Go from android device Pixel.
```



```
> Metro waiting on exp://192.168.0.22:19000
> Scan the QR code above with Expo Go (Android) or the Camera app (iOS)

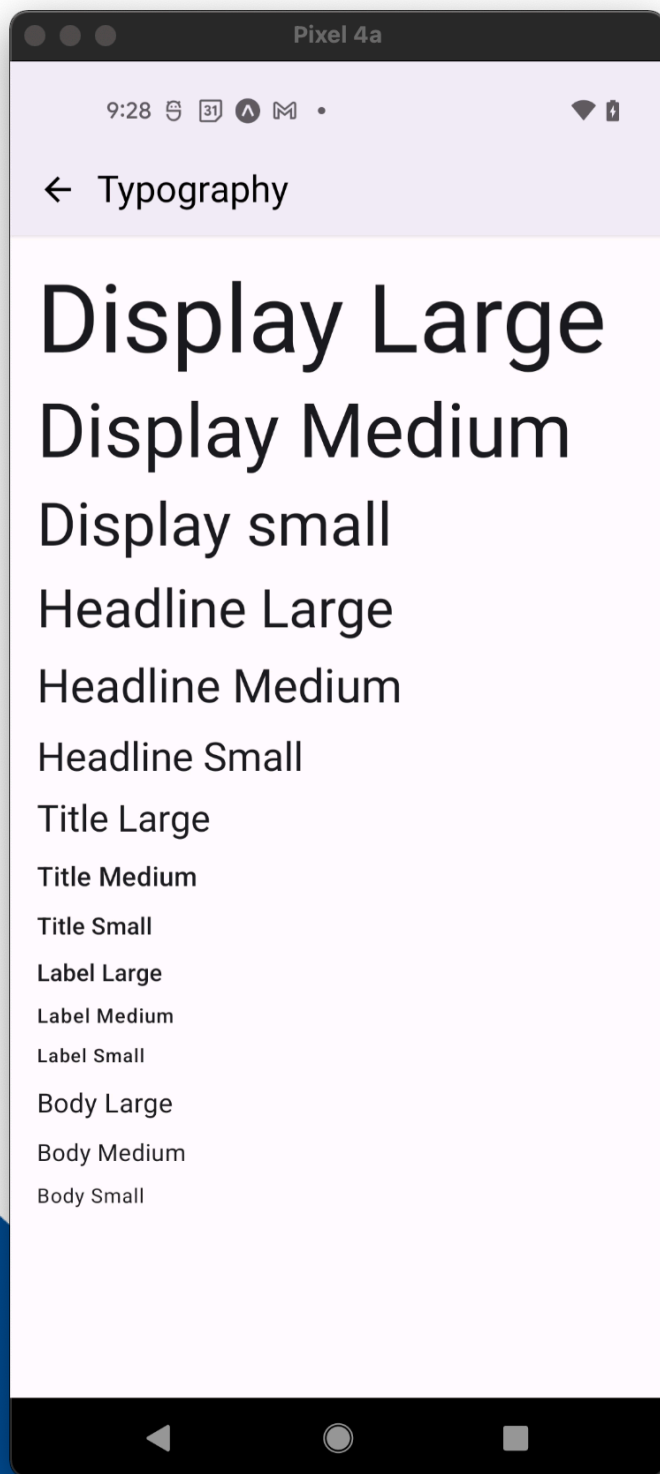
> Press a | open Android
> Press i | open iOS simulator
> Press w | open web

> 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.
```





```
import * as React from 'react';
import { Text } from 'react-native-paper';

const MyComponent = () => (
  <>
    <Text variant="displayLarge">Display Large</Text>
    <Text variant="displayMedium">Display Medium</Text>
    <Text variant="displaySmall">Display small</Text>

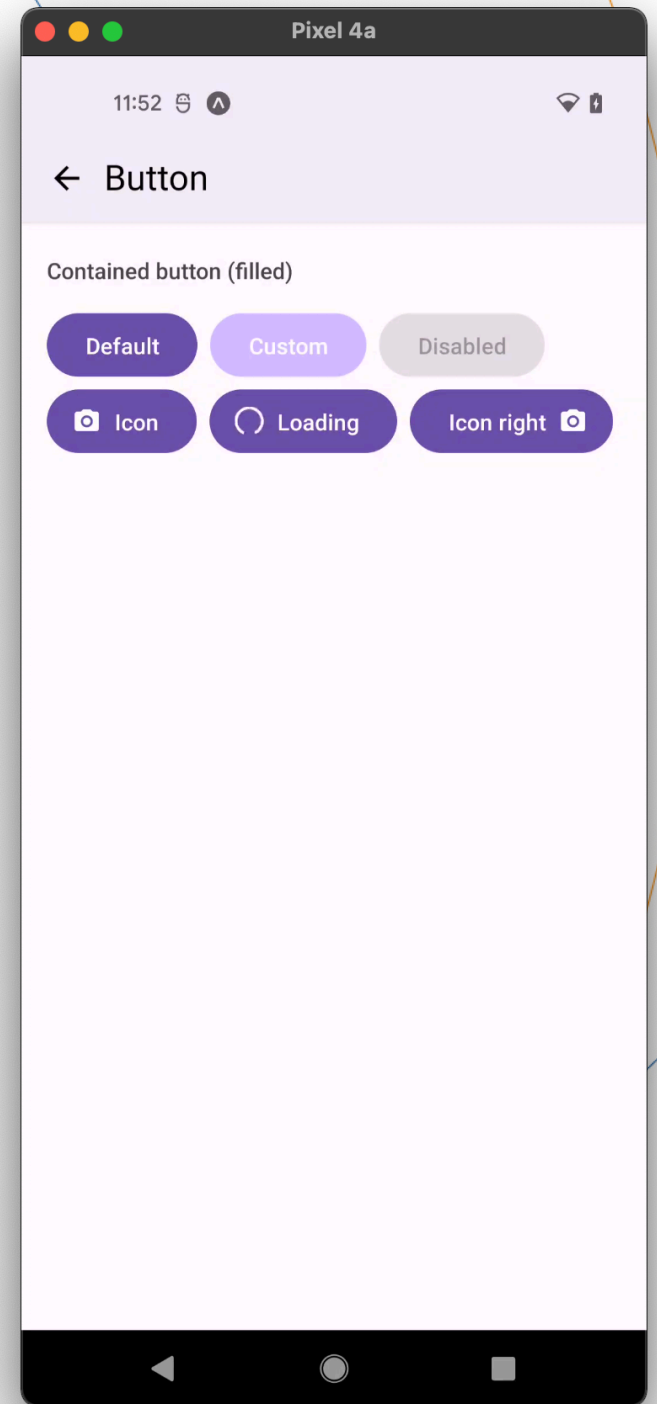
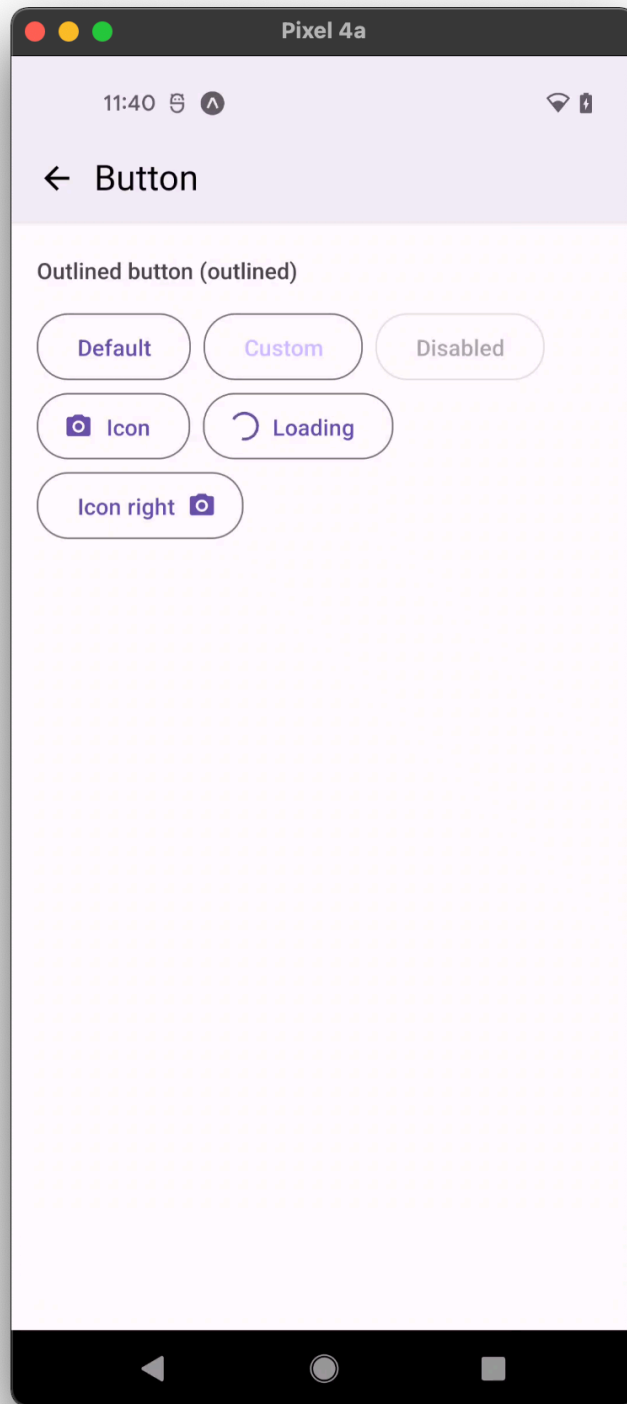
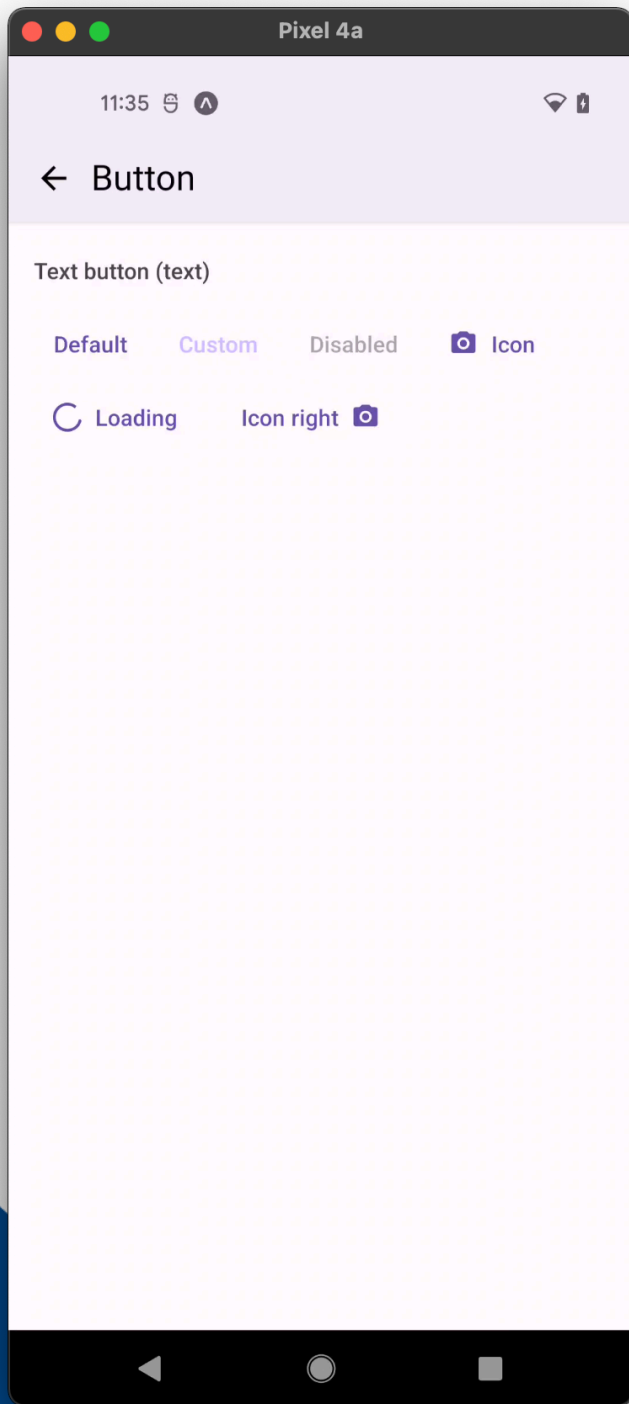
    <Text variant="headlineLarge">Headline Large</Text>
    <Text variant="headlineMedium">Headline Medium</Text>
    <Text variant="headlineSmall">Headline Small</Text>

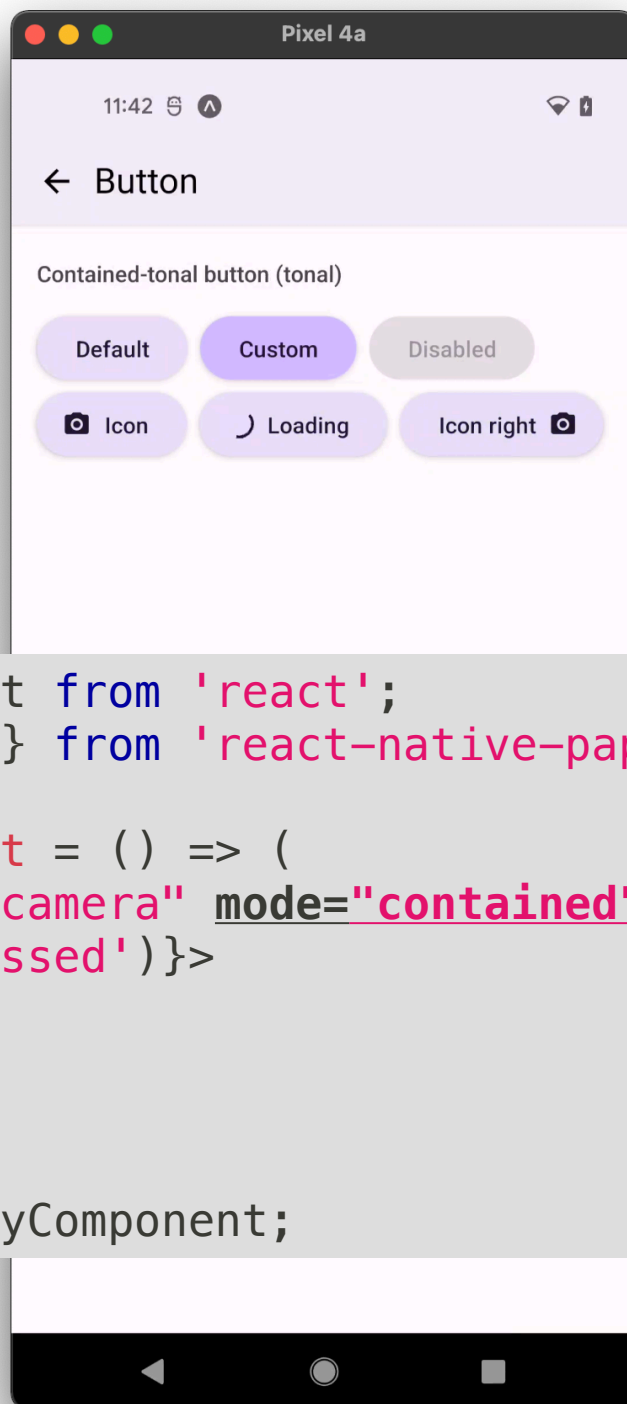
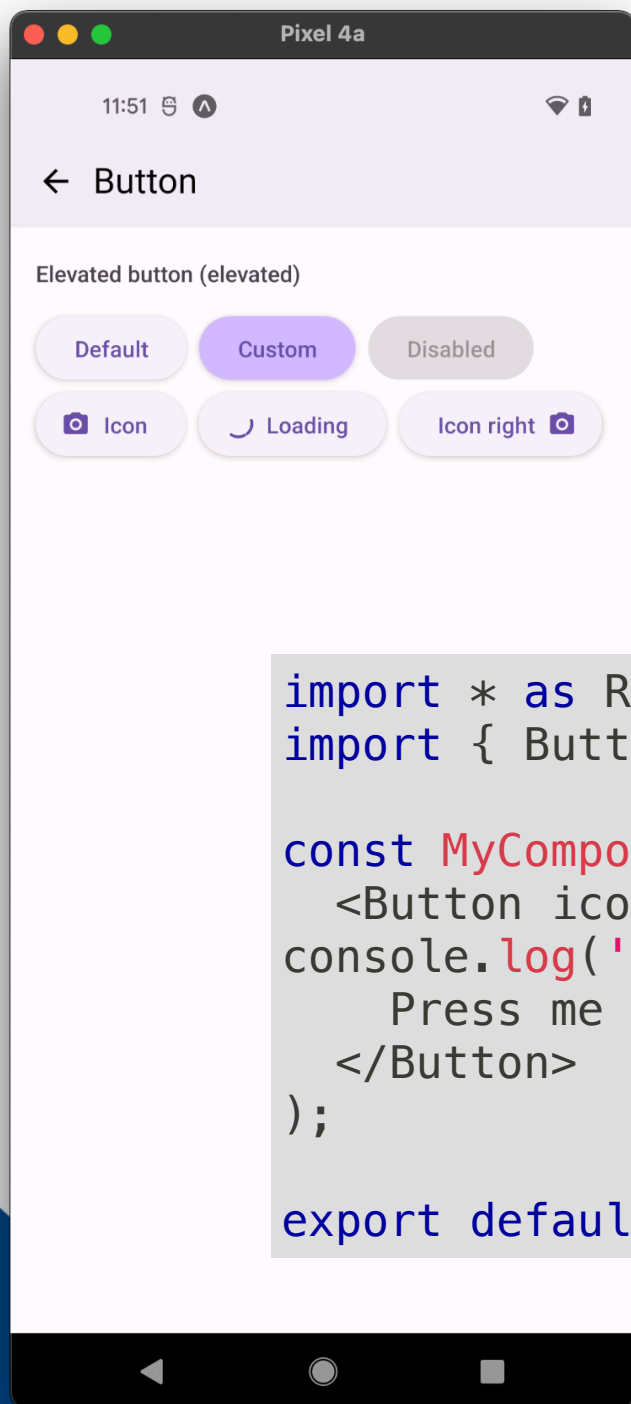
    <Text variant="titleLarge">Title Large</Text>
    <Text variant="titleMedium">Title Medium</Text>
    <Text variant="titleSmall">Title Small</Text>

    <Text variant="bodyLarge">Body Large</Text>
    <Text variant="bodyMedium">Body Medium</Text>
    <Text variant="bodySmall">Body Small</Text>

    <Text variant="labelLarge">Label Large</Text>
    <Text variant="labelMedium">Label Medium</Text>
    <Text variant="labelSmall">Label Small</Text>
  </>
);

export default MyComponent;
```



```
import * as React from 'react';
import { Button } from 'react-native-paper';

const MyComponent = () => (
  <Button icon="camera" mode="contained" onPress={() =>
    console.log('Pressed')}>
    Press me
  </Button>
);

export default MyComponent;
```

PROJETO INTEGRADOR

