



EVERARDO GUADALUPE/ *aloha!*

---

# REACT NATIVE

# CONTENIDO

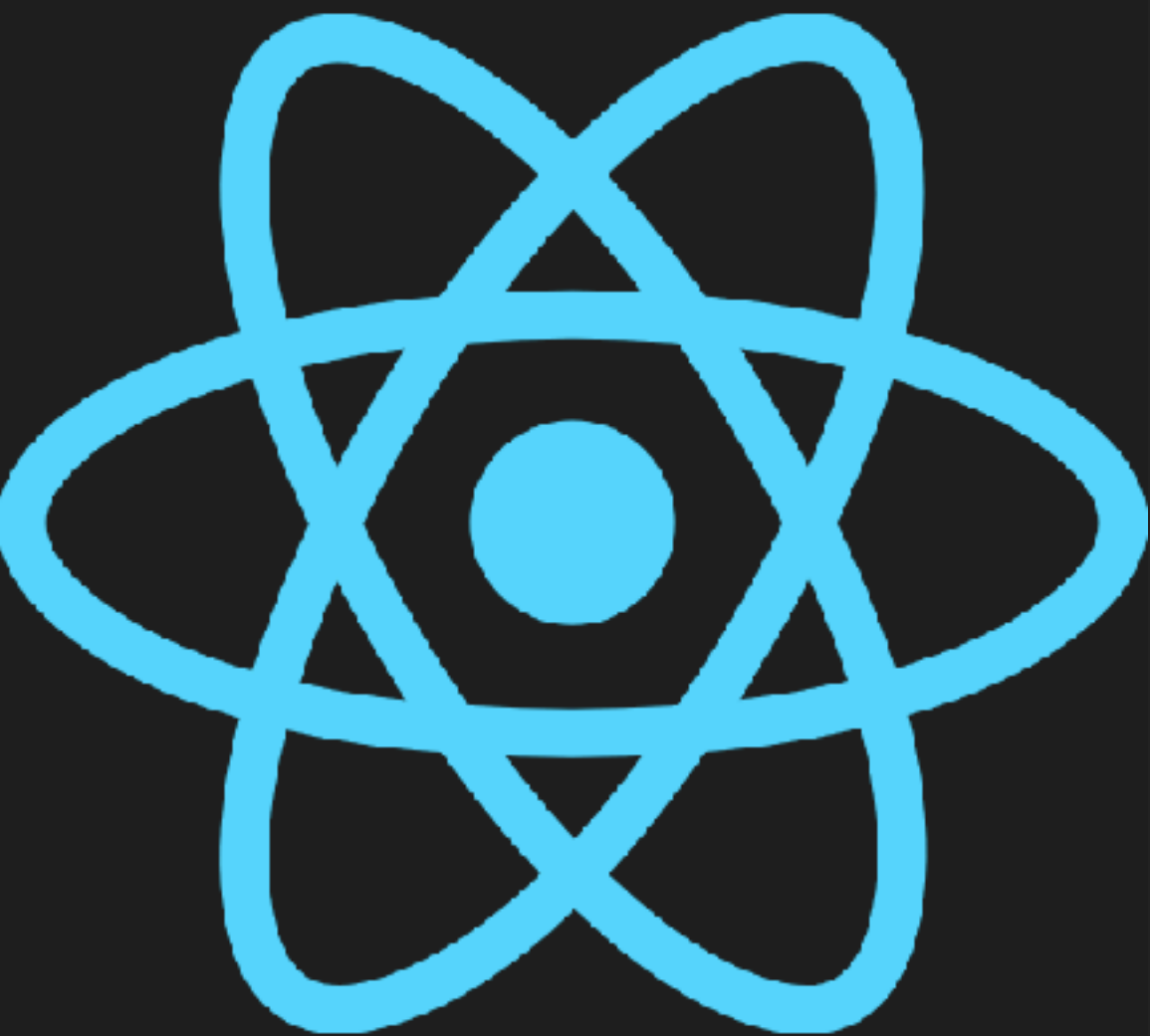
- ▶ Instalación
- ▶ ¿Facebook y React?
- ▶ React Native
- ▶ Ventajas
- ▶ Desventajas

# CONTENIDO

- ▶ Creando un proyecto - Estructura y Ejecución
- ▶ Componentes
- ▶ Creando la navegación
- ▶ Explorando componentes
  - ▶ Texto
  - ▶ Inputs
  - ▶ Contenedores
  - ▶ Alerts
  - ▶ Listas
  - ▶ Imagenes
  - ▶ Comunicación

## CONTENIDO

- ▶ Creando una App
- ▶ Crear una app de detalles
- ▶ Creando componentes
- ▶ Menu y los componentes



INICIO

---

**INSTALACIÓN**

# INSTALACIÓN

## ▶ Opción A:

- ▶ <https://facebook.github.io/react-native/docs/getting-started.html>
- ▶ Tutorial: <https://goo.gl/EwH1SU>

## ▶ Opción B

- ▶ Descargar e instalar Node JS (~ 12 MB)  
<https://nodejs.org/en/>
- ▶ Descargar las Librerías de React[Comando de terminal]  
npm install -g react-native-cli
- ▶ Development IDE (Visual Studio Code o Web Storm) (~34 MB - ~150 MB )  
<https://code.visualstudio.com/>  
<https://www.jetbrains.com/webstorm/>

## ¿COMO INICIAR?

▶ Creamos un nuevo: miProyecto

▶ Instrucciones:

```
react-native init miProyecto
```

```
cd miProyecto
```

```
react-native start
```

Compila el proyecto

▶ React-native run-android

▶ React-native run-ios

Inicia el proyecto (Android/iOS)

## NOTAS:

Para probar los proyectos se necesita:

- ▶ Para Windows
  - ▶ Android Studio
  - ▶ Android SDK (6.0)
  - ▶ Simulador
- ▶ Para MacOSX
  - ▶ Xcode
  - ▶ Simulador



## OPCIÓN B

Si no quieres instalar simuladores y demás cosas en tu PC usaremos:

- ▶ Desarrollo en vivo y checarlo en tu dispositivo
  - ▶ Entra a <https://snack.expo.io/>
  - ▶ Descarga e instala la App: Expo Client
  - ▶ Scanea el QR a traves de la App Expo.
  - ▶ Listo

## ERRORES COMUNES

### Unable to resolve module error

Algunas veces quedan los paquetes atorados por ende ocupamos limpiar el cache para poderlos obtener:

- ▶ `npm cache clean`
- ▶ `delete ./node_modules`
- ▶ `delete %temp%/react-*`
- ▶ `npm install`

## ERRORES COMUNES

error: bundling: NotFoundError: Cannot find entry file  
index.android.js

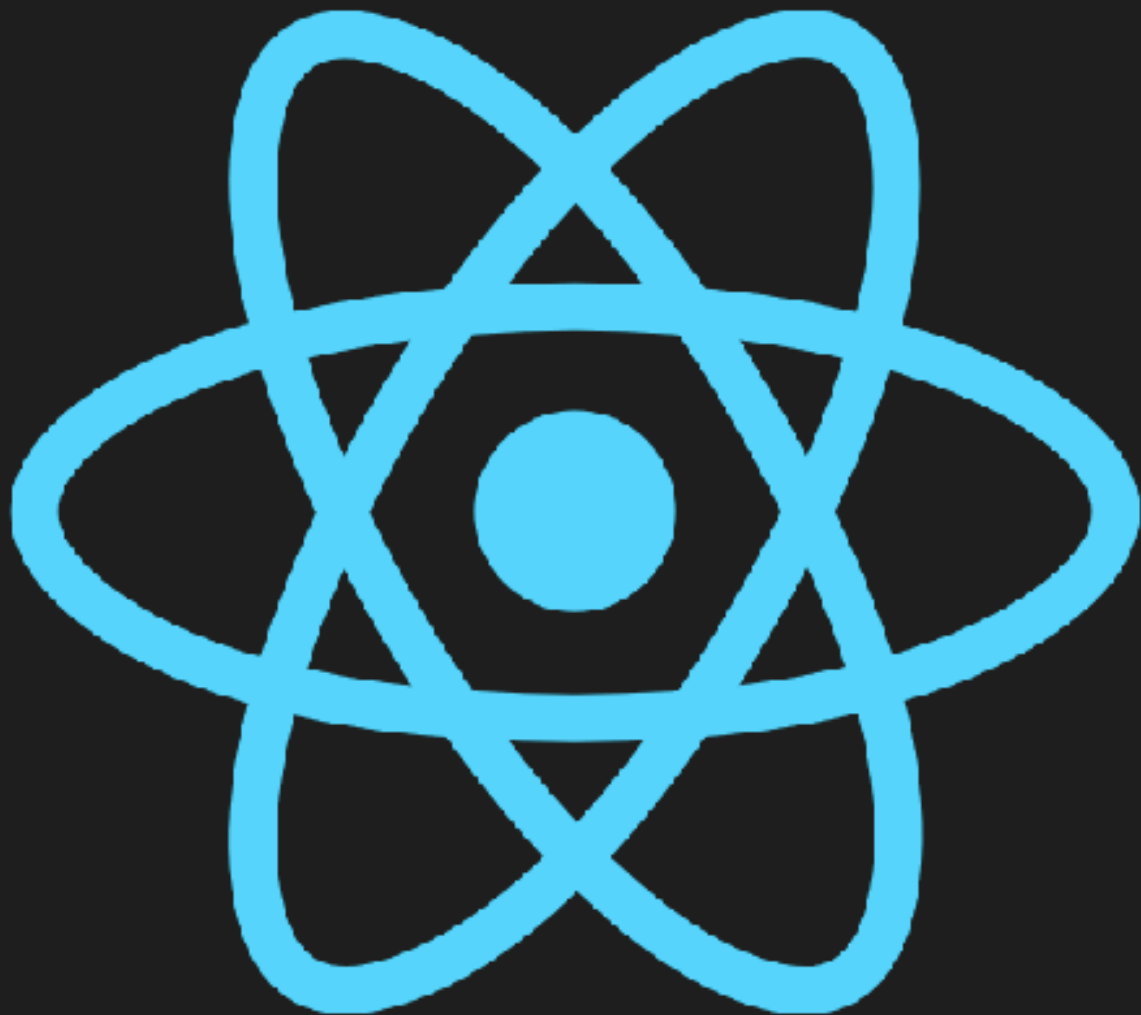
Ejecuta los siguientes comandos en orden para arreglarlo

- ▶ react-native start
- ▶ react-native run-android

## ERRORES COMUNES

Si requieres una nueva version usa:

- ▶ `npm uninstall -g react-native-cli`
- ▶ `npm install -g react-native-cli`
- ▶ `react-native init miProyecto`



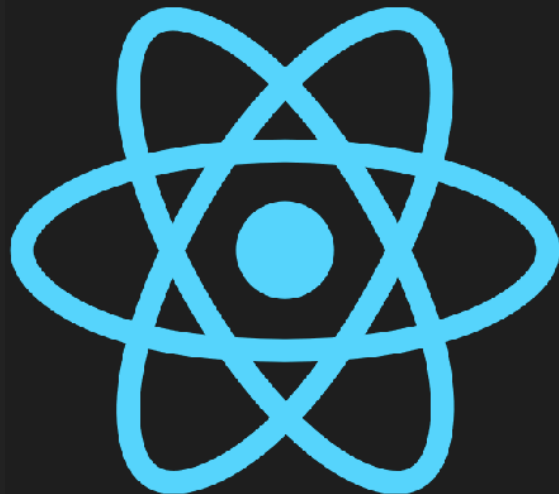
REACT NATIVE

---

¿FACEBOOK Y REACT?

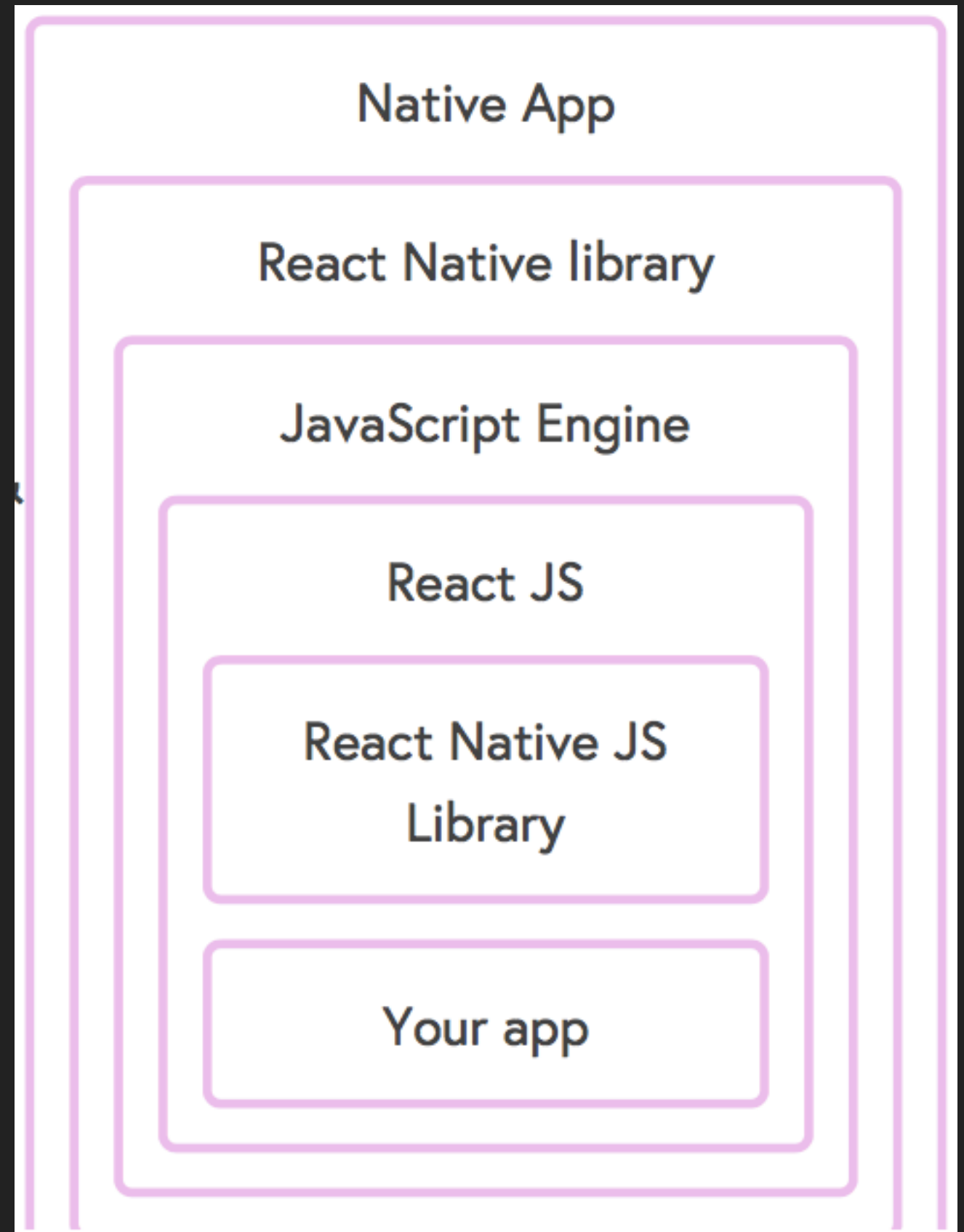
# ¿QUE ES?

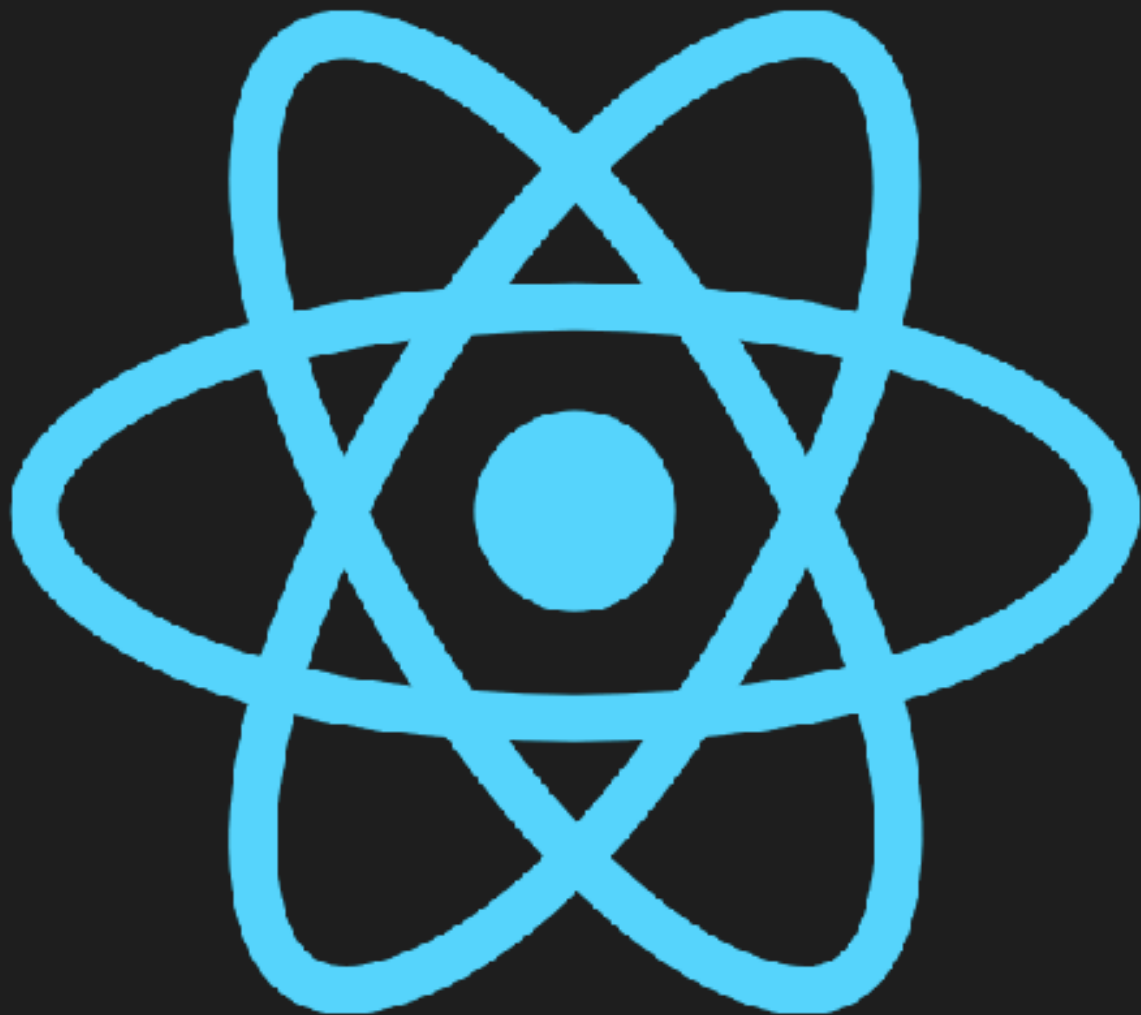
React Native es una de las tecnologías más de moda en este momento, y nadie puede negar que está revolucionando el desarrollo móvil.



# LA BASE

- ▶ **LIBRE**
- ▶ **GRAN COMUNIDAD**
- ▶ **FACILIDAD DE USO**





REACT NATIVE

---

**ASPECTOS BASICOS**



## REACT NATIVE – DATOS IMPORTANTES

- ▶ Usa ES6
- ▶ Para usar iOS requieres Xcode y MacOSX
- ▶ Para usar Android requieres Android studio
- ▶ React-Native no utiliza HTML para procesar la aplicación, pero proporciona componentes alternativos que funcionan de manera similar. Esos componentes de React-Native mapean los componentes reales de iOS nativos o de interfaz de usuario de Android que se procesan en la aplicación.

## ¿ECMAScript 6?

Cuando hablamos de ECMAScript, nos referimos al estándar que ya va por la versión ES6 y determina cómo emplear el lenguaje Javascript, que permite a los fabricantes de software desarrollar las herramientas adecuada para interpretarlo correctamente.

**ES6**

```
// Ejemplo 1
([param] [, param]) => {
  statements
}
```

```
// Ejemplo 2
param => expression
```

**ANTERIOR**

```
// Ejemplo 1
function ([param] [, param]) {
  statements
}
```

```
// Ejemplo 2
function (param) {
  return expression
}
```

# HTML VS COMPONENTES DE REACT NATIVE

HTML	React Native
<Div	<View
<P	<Text
<a href	<TouchableOpacity onPress=
<img src=	<Image source=
<input type=text onChange=	<TextInput onChangeText=
<input type=button />	<Button onPress = {handlePress} title = "Red button!" />
<select onChange=""><option value='red'>red<option>	<Picker onChange = {this.fn1} mode="dialog">

## EJEMPLO DE UN PROGRAMA EN REACT NATIVE

```
import React, { Component } from 'react';
import { View, Text } from 'react-native';
export default class App extends Component {
  render() {
    return (
      <View style={styles.container}>
        <Text style={styles.intro}>Hello world!</Text>
      </View>
    );
  }
}
```

## USANDO HOJAS DE ESTILO REACT NATIVE

Como lo declaras:

```
const styles = StyleSheet.create({  
  container: { flex: 1, },  
  content: { backgroundColor: '#fff', padding: 30, },  
  button: { alignSelf: 'center', marginTop: 20, width: 100, },  
  image: { width: 40, height: 40, marginRight: 6, borderRadius: 20 }  
  intro: { fontSize: 20, }  
});
```

Como usarlo:

```
<View style={styles.container} > Hola </View>
```

## TIPS: COMO USARLOS

### En linea:

```
<view style={{backgroundColor:'#ff0000'}}>  
  Hola  
</view>
```

### En linea + Estilo normal

```
<view style={[styles.container, {color: 'white', backgroundColor: '#333'}]} >  
  Hola  
</view>
```

### Dos estilos

```
<view style={[ styles.container , styles.style1 ]} >  
  Hola  
</view>
```

### Depende de la plataforma

```
{marginTop: Platform.OS === 'android' ? 22 : 0}
```

# FLEXBOX

Flexbox es un sistema de elementos flexibles que llega con la idea de olvidar estos mecanismos y acostumbrarnos a una mecánica más potente, limpia y personalizable, en la que los elementos HTML se adaptan y colocan automáticamente y es más fácil personalizar los diseños.

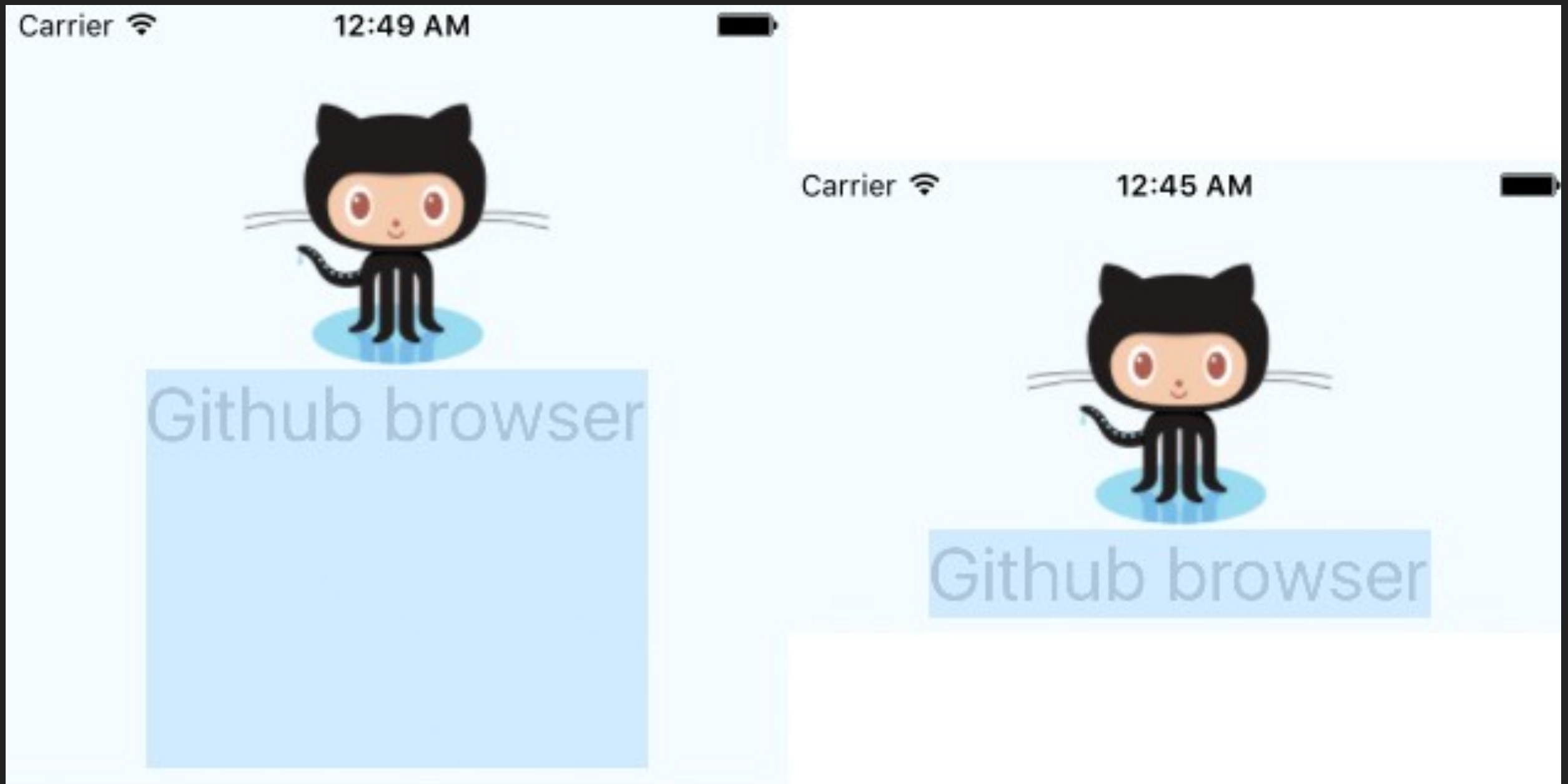
Recursos:

<https://goo.gl/gUJVRm>

<https://lenguajecss.com/p/css/propiedades/flexbox>

<https://www.udemy.com/curso-de-flexbox-desde-0/>

**FLEX : FLEX:1 => EL CONTENEDOR SE EXPANDE A TODO EL ANCHO**



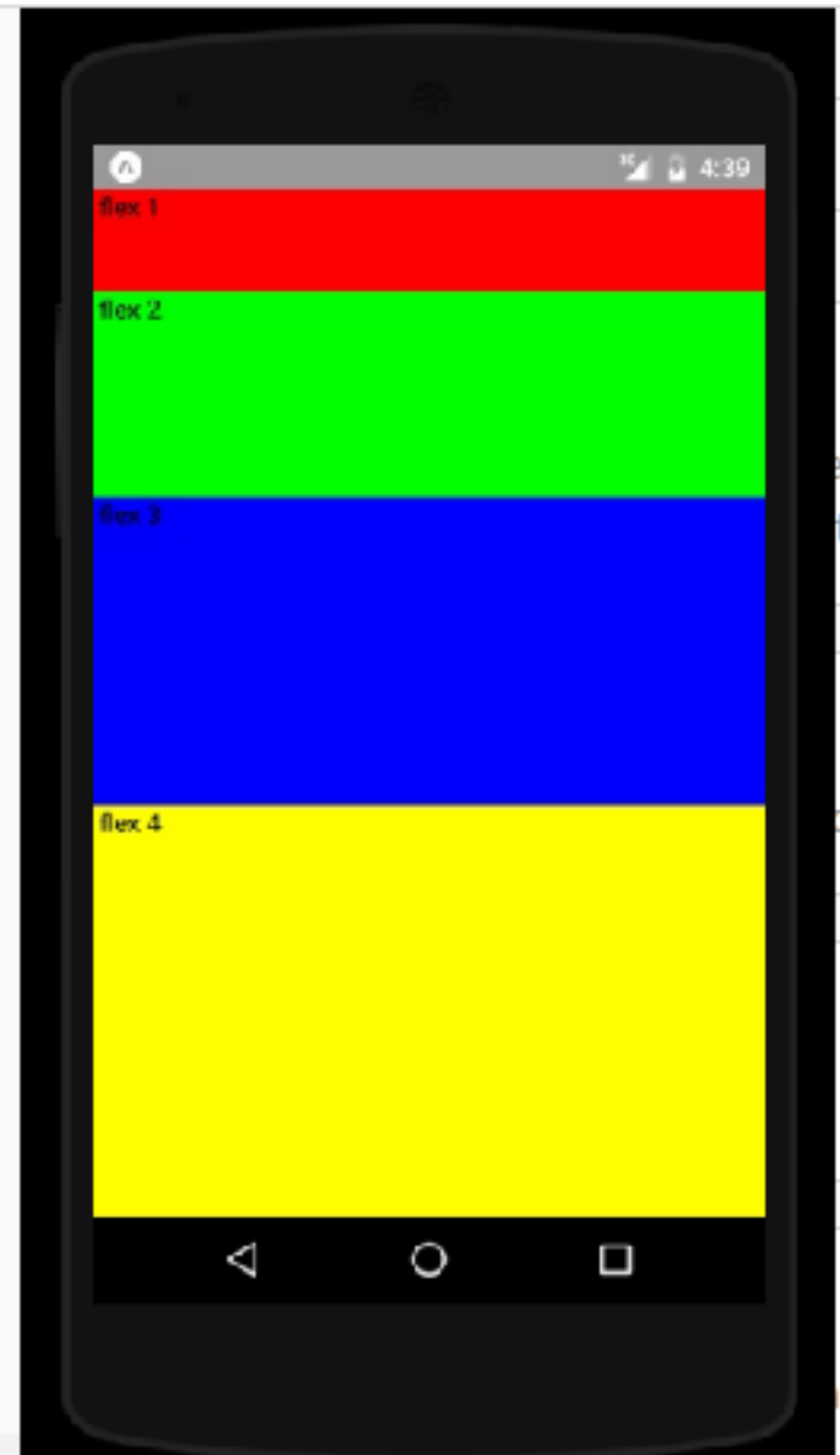
**CON FLEX: 1**

**SIN FLEX**



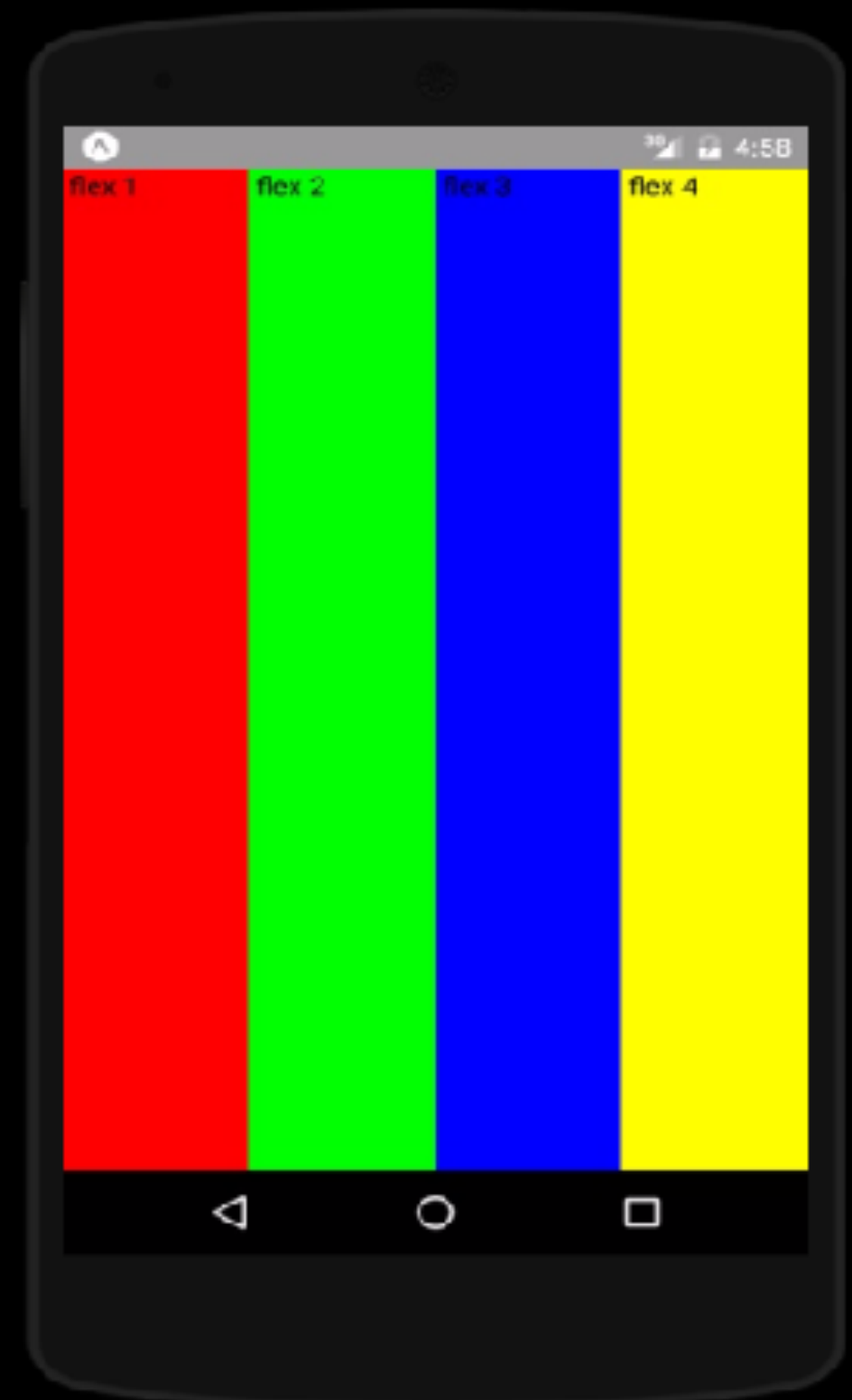
# FLEX : EN RELACION A OTROS FLEX

```
1 import React, { Component } from 'react';
2 import { Text, View, StyleSheet } from 'react-native';
3 import { Constants } from 'expo';
4
5 export default class App extends Component {
6   render() {
7     return (
8       <View style={styles.container}>
9         <Text style={styles.paragraph1}> flex 1    </Text>
10        <Text style={styles.paragraph2}> flex 2    </Text>
11        <Text style={styles.paragraph3}> flex 3    </Text>
12        <Text style={styles.paragraph4}> flex 4    </Text>
13      </View>
14    );
15  }
16 }
17
18 const styles = StyleSheet.create({
19   container: {flex: 1,paddingTop: Constants.statusBarHeight,},
20   paragraph1: { flex: 1,backgroundColor: '#ff0000',},
21   paragraph2: { flex: 2,backgroundColor: '#00ff00',},
22   paragraph3: { flex: 3,backgroundColor: '#0000ff',},
23   paragraph4: { flex: 4,backgroundColor: '#ffff00',},
24 });
25
```

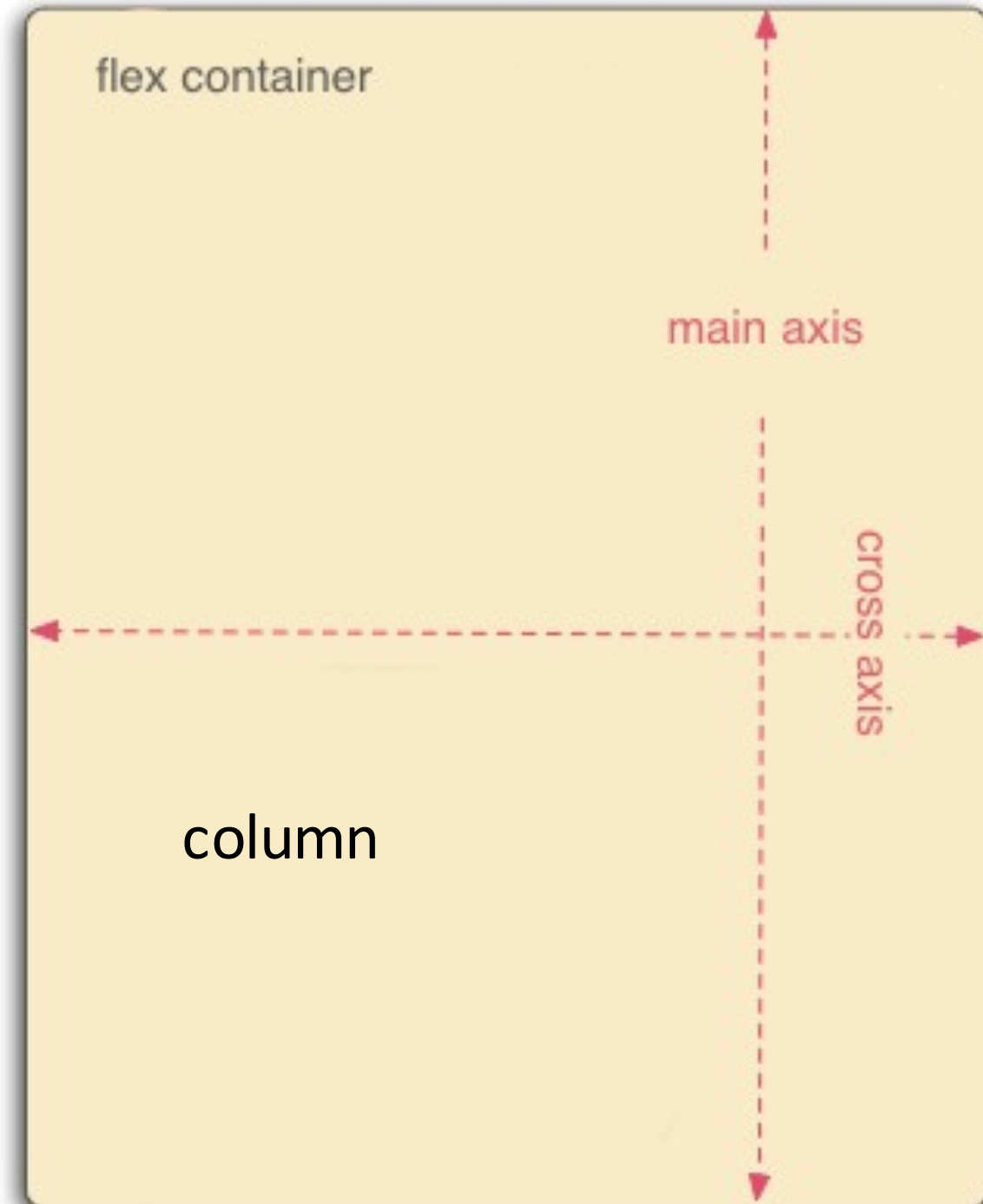
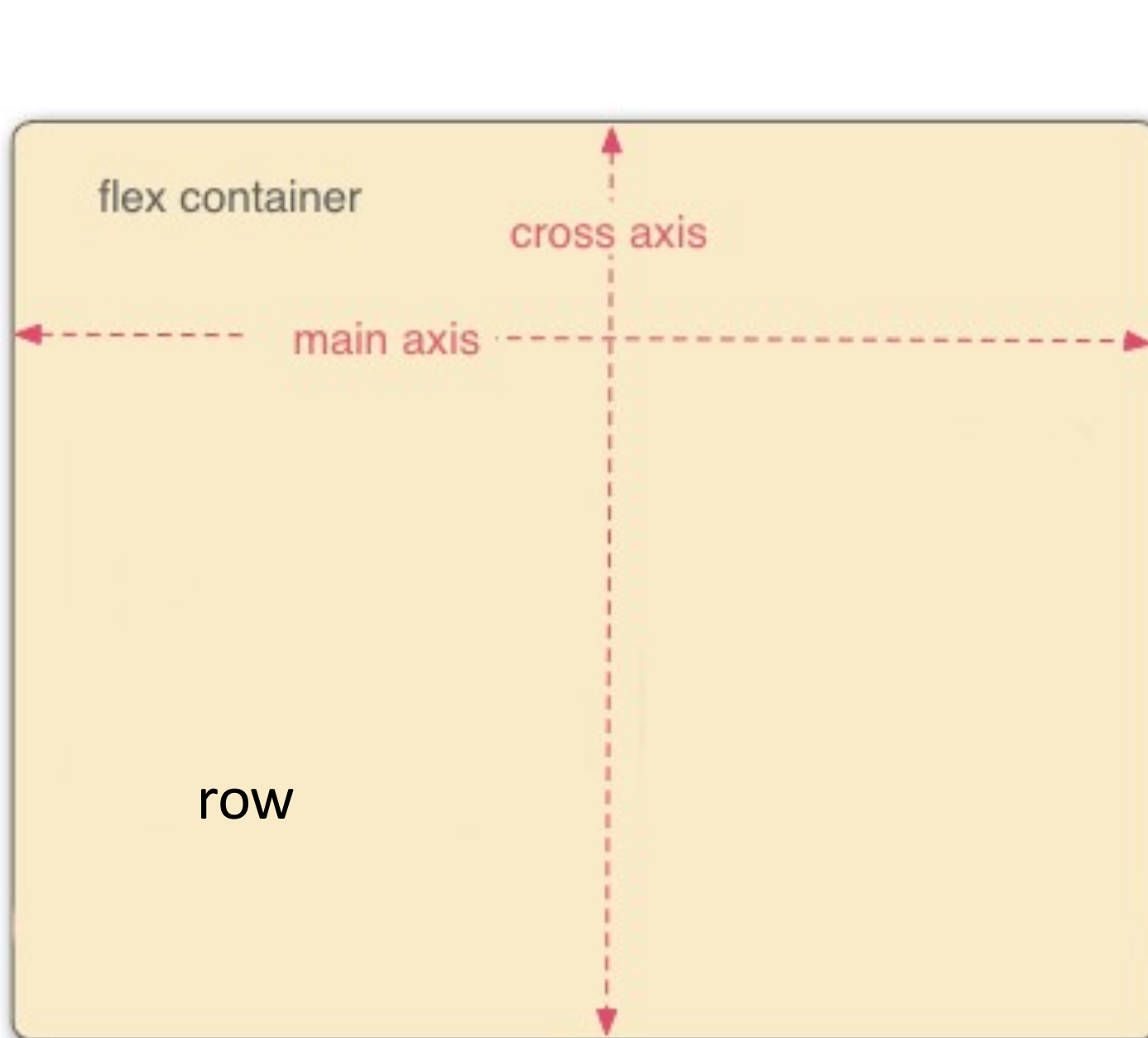


# FLEX : DIRECTION

```
1 import React, { Component } from 'react';
2 import { Text, View } from 'react-native';
3 import { Constants } from 'expo';
4
5 export default class App extends Component {
6   render() {
7     return (
8       <View style={styles.container}>
9         <Text style={styles.paragraph1}> flex 1 </Text>
10        <Text style={styles.paragraph2}> flex 2 </Text>
11        <Text style={styles.paragraph3}> flex 3 </Text>
12        <Text style={styles.paragraph4}> flex 4 </Text>
13      </View>
14    );
15  }
16 }
17
18 const styles = {
19   container: {flexDirection:'row' ,flex: 1,
20   paddingTop: Constants.statusBarHeight,},
21
22   paragraph1: { flex: 1,backgroundColor: '#ff0000',},
23   paragraph2: { flex: 1,backgroundColor: '#00ff00',},
24   paragraph3: { flex: 1,backgroundColor: '#0000ff',},
25   paragraph4: { flex: 1,backgroundColor: '#ffff00',},
```



## FLEX : PRINCIPALES EJES



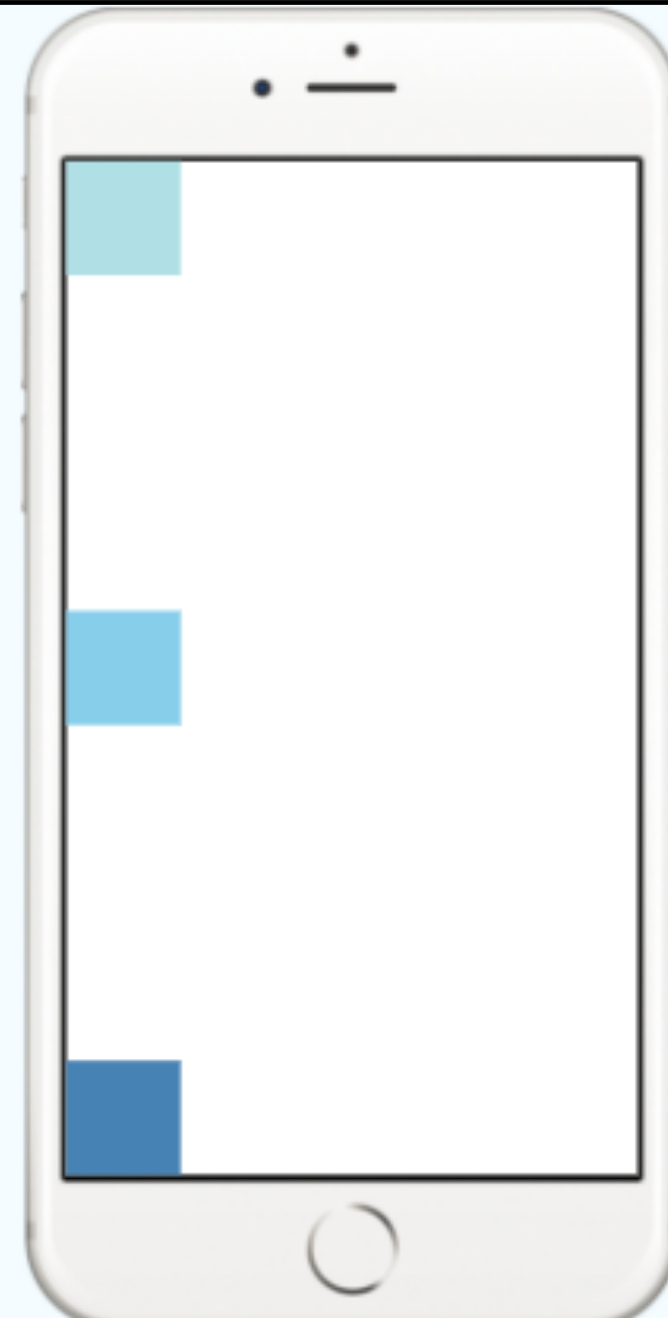
# FLEX : JUSTIFYCONTENT

FLEX-START, CENTER, FLEX-END, SPACE-AROUND, AND SPACE-BETWEEN.

```
1 import React, { Component } from 'react';
2 import { AppRegistry, View } from 'react-native';
3
4 export default class JustifyContentBasics extends Component {
5   render() {
6     return (
7       // Try setting `justifyContent` to `center`.
8       // Try setting `flexDirection` to `row`.
9       <View style={{
10         flex: 1,
11         flexDirection: 'column',
12         justifyContent: 'space-between',
13       }}>
14         <View style={{width: 50, height: 50, backgroundColor: 'powderblue'}}
15         <View style={{width: 50, height: 50, backgroundColor: 'skyblue'}} /
16         <View style={{width: 50, height: 50, backgroundColor: 'steelblue'}}
17       </View>
18     );
19   }
20 }
```

No Errors

Show Details



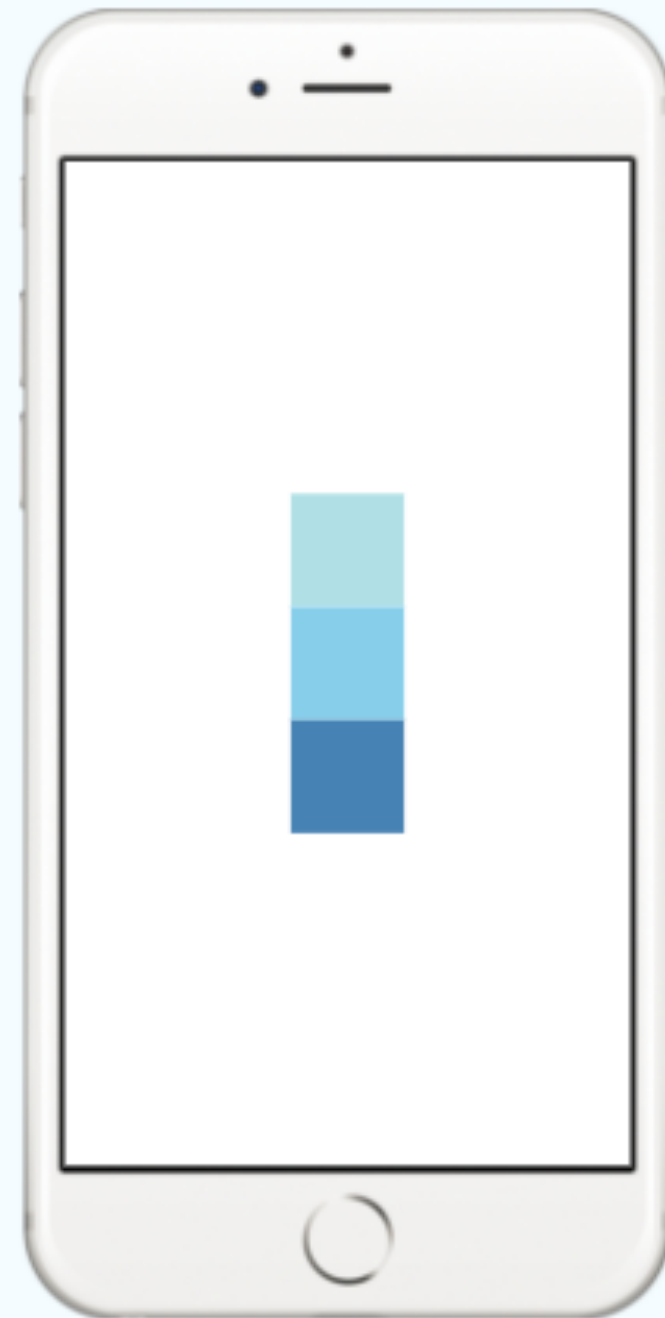
# FLEX : ALIGNITEMS

FLEX-START, CENTER, FLEX-END, AND STRETCH.

```
7 // Try setting `alignItems` to 'flex-start'
8 // Try setting `justifyContent` to 'flex-end'.
9 // Try setting `flexDirection` to 'row'.
10 <View style={{
11   flex: 1,
12   flexDirection: 'column',
13   justifyContent: 'center',
14   alignItems: 'center',
15 }}>
16   <View style={{width: 50, height: 50, backgroundColor: 'powderblue'}}
17   <View style={{width: 50, height: 50, backgroundColor: 'skyblue'}} /
18   <View style={{width: 50, height: 50, backgroundColor: 'steelblue'}}
19 </View>
20 );
21 }
22 };
23
24 // skip this line if using Create React Native App
25
```

No Errors

Show Details





# UI Component Library

---

Activity Indicator

Picker

Touchable

ActionSheet

Date Picker

ScrollView

TouchableOpacity

Alert

Image

Slider

Touchable  
Highlight

Border

ListView

TabBar

Touchable  
WithoutFeedback

StatusBarIOS

MapView

Text

View

StatusBarIOS

Navigator

TextInput

WebView

CameraRoll

# REACT INTERNALS



## View **lifecycle**

MOUNT

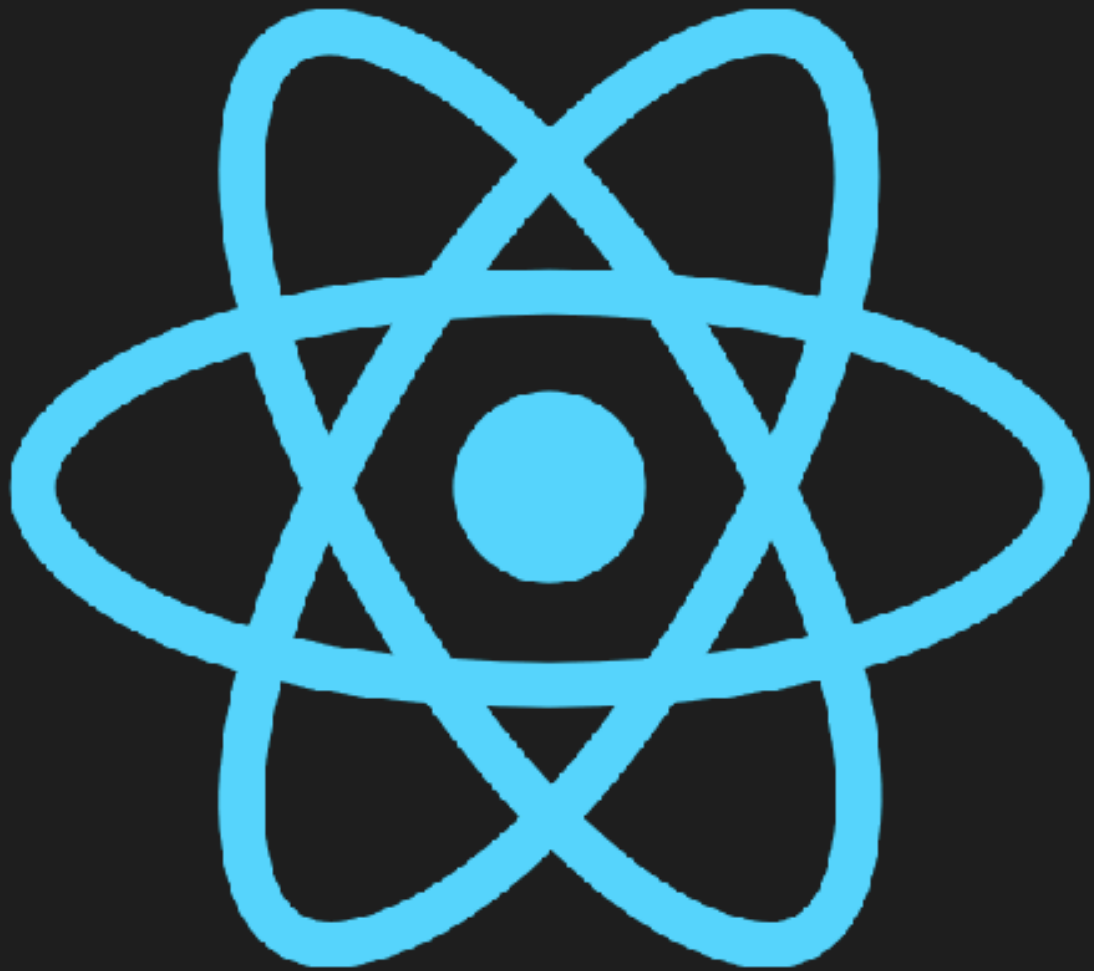
- componentWillMount
- render
- componentDidMount

UPDATE

- shouldUpdateComponent
- componentWillUpdate
- render
- componentDidUpdate

UNMOUNT

- componentWillUnmount



CREANDO UN  
PROYECTO

---

**ESTRUCTURA Y  
EJECUCIÓN**



## CREAR UN PROYECTO

- ▶ En caso del emulador
  - ▶ `react-native init miProyecto`
  - ▶ `Cd miProyecto`
  - ▶ Para iOS emulador: `react-native run-ios`
  - ▶ Para Android emulador: `react-native run-android`

## OPCIÓN B

Ahora ejecutamos nuestro proyecto, con los siguientes comandos de consola.

- ▶ `npm i -g create-react-native-app`
- ▶ `create-react-native-app miProyecto`
- ▶ `cd miProyecto`
- ▶ `npm start`

Ahora solamente lee el código QR usando la App.

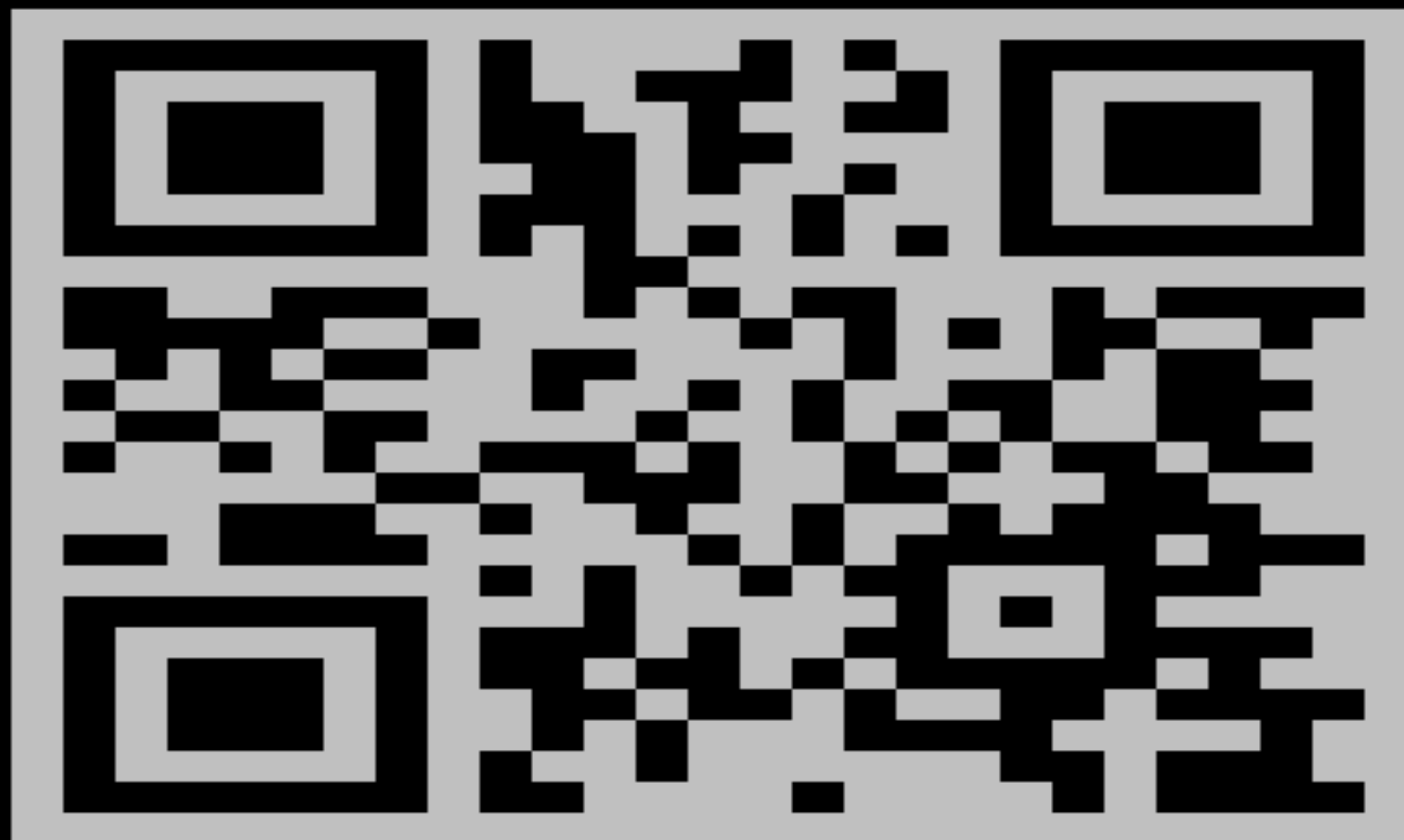
```
C:\Users\Administrator\project2>npm start
```

```
> project2@0.1.0 start C:\Users\Administrator\project2
> react-native-scripts start
```

```
13:00:37: Starting packager...
```

```
Packager started!
```

To view your app with live reloading, point the Expo app to this QR code.  
You'll find the QR scanner on the Projects tab of the app.



Or enter this address in the Expo app's search bar:

```
exp://9.164.167.94:19000
```

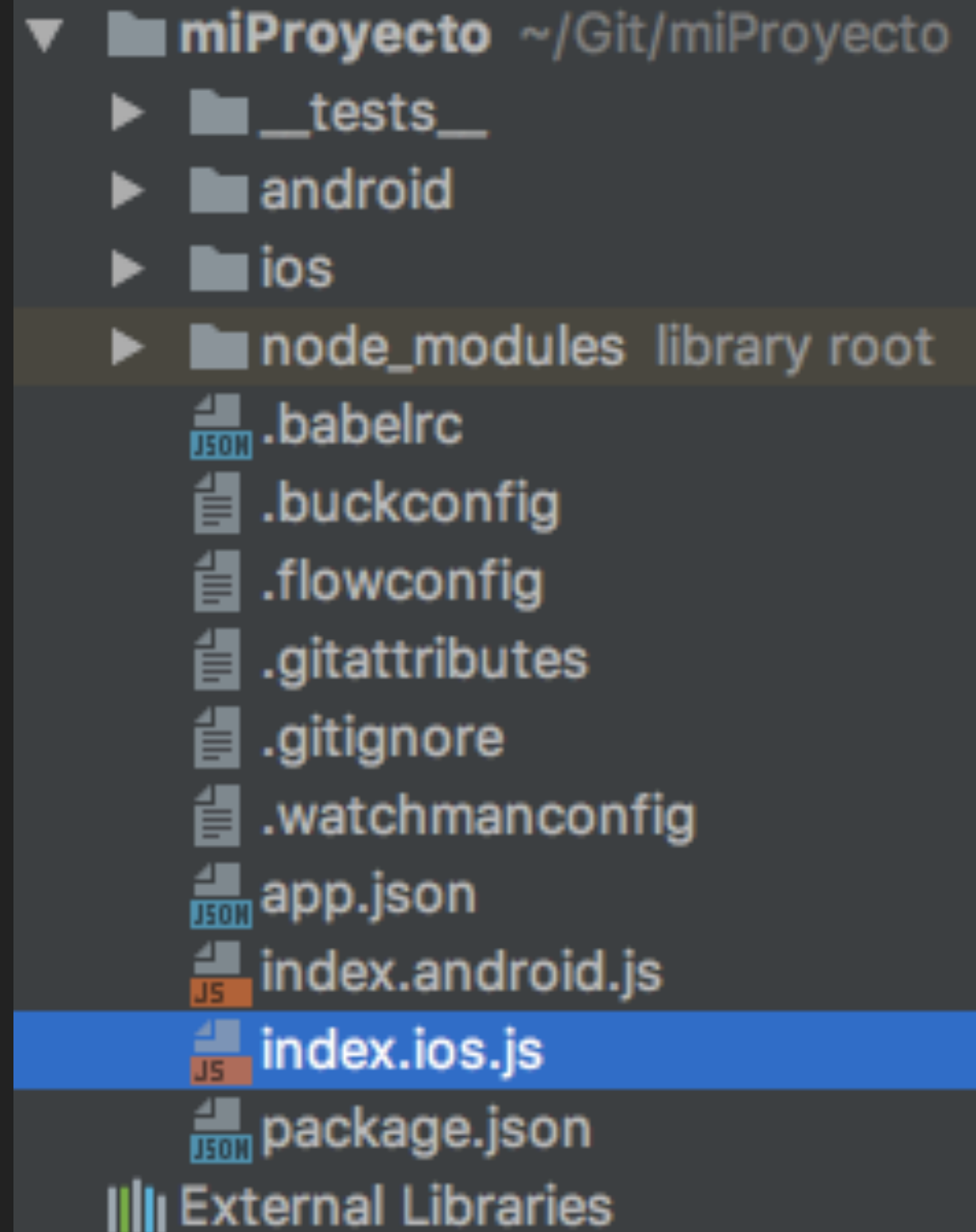
Your phone will need to be on the same local network as this computer.

For links to install the Expo app, please visit <https://expo.io>.

Logs from serving your app will appear here. Press Ctrl+C at any time to stop.

# ESTRUCTURA

- ▶ General
  - ▶ Android
  - ▶ iOS
- ▶ App
- ▶ No siempre es lo mismo.



## EJECUCIÓN – HERRAMIENTAS FUNDAMENTALES

- ▶ Reload
- ▶ Remote js Debug
- ▶ Enable Live Reload
- ▶ Actualizar el código en tiempo real

## CREANDO UN COMPONENTE

- ▶ ¿Que es?
- ▶ Componente Inicio
  - ▶ Crear una carpeta llamada "app"
  - ▶ Crear una carpeta llamada "componentes"
  - ▶ Crear un archivo llamado "ComponenteInicio.js"
- ▶ <https://goo.gl/9YgnVY>

## CREANDO UN MENU

Para crear un menu ocupamos lo siguiente:

- ▶ `npm i react-native-deprecated-custom-components`
- ▶ `npm install react-native-side-menu --save`

Estas instrucciones se necesitan teclear en la Terminal o el CMD.

## CREANDO UN MENU

Ahora necesitamos crear el menu.

- ▶ En la carpeta de componentes, creamos un archivo llamado "Menu" con la extensión JS(JavaScript).
- ▶ Y programamos.
- ▶ <https://goo.gl/y44Lhs>



## AÑADIENDO COMPONENTES

Ahora vamos a añadir componentes que nos ayudaran a comprender como funciona React Native.

- ▶ Texto
- ▶ Imagen
- ▶ Inputs
- ▶ Contacto
- ▶ Lector RSS

# AÑADIENDO COMPONENTES

[HTTPS://GITHUB.COM/CLEVERG/REACTNATIVE](https://github.com/cleverg/reactnative)

## RECURSOS

- ▶ <http://alohacreativos.com/feed/>
- ▶ `npm install react-native-communications`
- ▶ <https://facebook.github.io/react-native/releases/0.21/docs/signed-apk-android.html>