	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN	
CICLO: 02	GUIA DE LABORATORIO #06	
	Nombre de la Práctica:	React Native Layout and Flexbox
	MATERIA:	Diseño y Programación de Software Multiplataforma

I. RESULTADOS DE APRENDIZAJE

Que el estudiante:

- Adquiera el dominio para trabajar con FlexDirection.
- Cree proyectos react native utilizando FlexDirection

II. INTRODUCCIÓN

FlexDirection agrega el estilo al componente en un eje principal de su diseño. Tiene una fila de propiedades y una columna para organizar a los niños horizontal y verticalmente respectivamente. La flexDirection predeterminada es una columna.

React Native Flexbox es un algoritmo para especificar el diseño de los hijos del componente. Proporciona un diseño uniforme en diferentes tamaños de pantalla.

Propiedad	Valor	Descripción
flexDirection	'column', 'row'	Se utiliza para alinear sus elementos vertical u horizontalmente.
justifyContent	'center', 'flex-start', 'flex-end', 'space-around', 'space-between'	Se utiliza para distribuir los elementos dentro del contenedor.
alignItems	'center', 'flex-start', 'flex-end', 'stretched'	Se utiliza para distribuir el elemento dentro del contenedor a lo largo del eje secundario (opuesto a flexDirection).

III. PROCEDIMIENTO DE GUÍA

PARTE I:

1. Cree un proyecto react-native con nombre agencia de viaje.
2. Para entender las propiedades del Flex vamos a digitar el siguiente código en nuestro archivo app.js

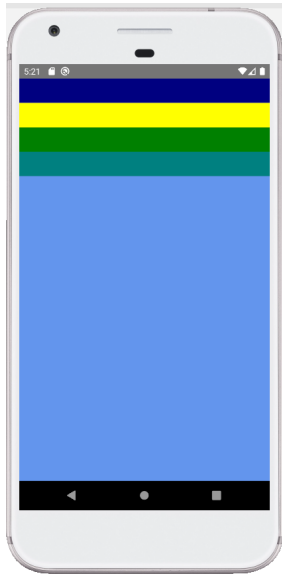
```
import React from 'react';
import {View, StyleSheet} from 'react-native';

const App = () =>{
  return(
    <>
    <View style={styles.contenedor}>
      <View style={styles.caja1}></View>
      <View style={styles.caja2}></View>
      <View style={styles.caja3}></View>
      <View style={styles.caja4}></View>
    </View>
    </>
  );
};

const styles = StyleSheet.create({
  contenedor:{
    backgroundColor:'cornflowerblue',
    flex:1,
    flexDirection:'column'
  },
  caja1:{
    padding:20,
    backgroundColor:'navy'
  },
  caja2:{
    padding:20,
    backgroundColor:'yellow'
  },
  caja3:{
    padding:20,
    backgroundColor:'green'
  },
  caja4:{
    padding:20,
    backgroundColor:'teal'
  },
});

export default App;
```

3. Este código de ejemplo deberá mostrar lo siguiente:



4. ahora vamos a modificar la siguiente propiedad en nuestra constante estilo para ver las cajas de izquierda a derecha:

```
const styles = StyleSheet.create({
  contenedor: {
    backgroundColor: 'cornflowerblue',
    flex: 1,
    flexDirection: 'row'
  },
  ...
});
```

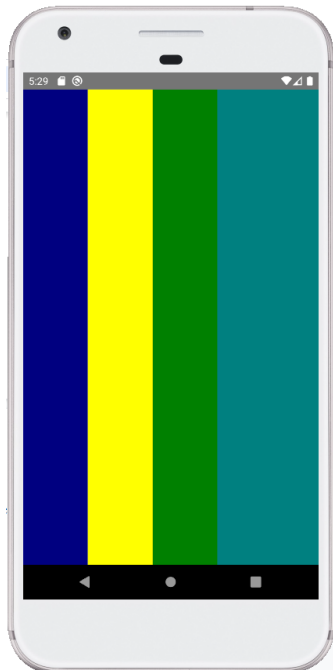
5. Este cambio debería presentar los elementos de la siguiente forma:



6. Ahora vamos a modificar el factor de crecimiento de cada uno de los elementos modifique el estilo de la siguiente forma:

```
const styles = StyleSheet.create({
  contenedor:{
    backgroundColor: 'cornflowerblue',
    flex:1,
    flexDirection: 'row'
  },
  caja1:{
    padding:20,
    backgroundColor: 'navy',
    flex:1,
  },
  caja2:{
    padding:20,
    backgroundColor: 'yellow',
    flex:1,
  },
  caja3:{
    padding:20,
    backgroundColor: 'green',
    flex:1,
  },
  caja4:{
    padding:20,
    backgroundColor: 'teal',
    flex:2,
  },
});
```

7. deberá de presentar la pantalla de la siguiente forma:



8. modifique las propiedades de estilo para probar la alineación de contenido, los valores que puede utilizar para esas propiedades las encuentra en la tabla presentada en la introducción pruebe todas las propiedades para que vea sus respectivos cambios en el diseño:

```
const styles = StyleSheet.create({
  contenedor:{
    backgroundColor: 'cornflowerblue',
    flex:1,
    flexDirection: 'row',
    justifyContent: 'space-around',
    alignItems: 'flex-end'
  },
});
```

PARTE II:

1. Ahora que hemos comprendido las propiedades de flex vamos a diseñar nuestra App para una agencia de viaje, primero descargue las imagenes en los recursos proporcionados, guardelas en una carpeta llamada src.
2. Ya con las imagenes agregadas a nuestro proyecto deberemos modificar nuestro App.js

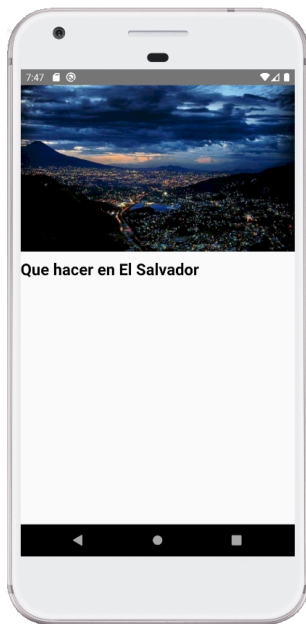
```
import React from 'react';
import {View, StyleSheet, Image} from 'react-native';

const App = () =>{
  return(
    <>
    <View >
      <View style={{flexDirection: 'row'}} >
        <Image
          style={styles.banner}
          source={require('./src/img/bg.jpg')} />
        </View>
      </View>
    </>
  );
};

const styles = StyleSheet.create({
  banner:{
    height:250,
    flex:1
  }
});

export default App;
```

3. obtendremos la siguiente vista:



4. crearemos un scrollview para mostrar actividades en El Salvador

- importar en las librerías el elemento ScrollView

```
import React from 'react';  
import {View, StyleSheet, Image, Text, ScrollView from 'react-native';
```

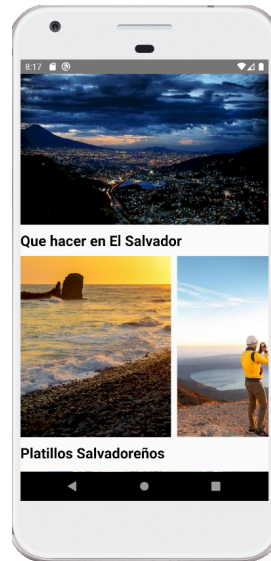
- agregaremos el siguiente código :

```
<view >  
  <View style={{flexDirection:'row'}} >  
    <Image  
      style={styles.banner}  
      source={require('./src/img/bg.jpg')} />  
  </View>  
  
  <View styles={styles.contenedor}>  
    <Text style={styles.titulo}>Que hacer en El Salvador</Text>  
    <ScrollView horizontal>  
      <View>  
        <Image  
          style={styles.ciudad}  
          source={require('./src/img/actividad1.jpg')} />  
      </View>  
      <View>  
        <Image  
          style={styles.ciudad}  
          source={require('./src/img/actividad2.jpg')} />  
      </View>  
      <View>  
        <Image  
          style={styles.ciudad}  
          source={require('./src/img/actividad3.jpg')} />  
      </View>  
      <View>  
        <Image  
          style={styles.ciudad}  
          source={require('./src/img/actividad4.jpg')} />  
      </View>  
      <View>  
        <Image  
          style={styles.ciudad}  
          source={require('./src/img/actividad5.jpg')} />  
      </View>  
    </ScrollView>  
  </View>  
</view>
```

- modificaremos los estilos:

```
const styles = StyleSheet.create({
  banner: {
    height: 250,
    flex: 1
  },
  titulo: {
    fontWeight: 'bold',
    fontSize: 24,
    marginVertical: 10
  },
  contenedor: {
    marginHorizontal: 10,
  },
  ciudad: {
    width: 250,
    height: 300,
    marginRight: 10
  }
});

export default App;
```



5. Ahora agregaremos una lista de platillos salvadoreños

- sustituir el View por un ScrollView

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

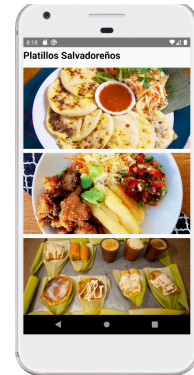
const App = () => {
  return (
    <ScrollView>
      <View style={{flexDirection: 'row'}}>
        <Image
          style={styles.banner}
          source={require('./src/img/bg.jpg')} />
      </View>
    </ScrollView>
  );
};
```

- agregar el siguiente código después del scrollview de cosas que hacer en El Salvador:

```
<Text style={styles.titulo}>Platillos Salvadoreños</Text>
<View>
  <View>
    <Image
      style={styles.mejores}
      source={require('./src/img/mejores1.jpg')} />
    </View>
    <View>
      <Image
        style={styles.mejores}
        source={require('./src/img/mejores2.jpg')} />
      </View>
    <View>
      <Image
        style={styles.mejores}
        source={require('./src/img/mejores3.jpg')} />
      </View>
    </View>
  </View>
```

- agregar la siguiente regla de estilo en styles

```
mejores:{
  width:'100%',
  height:200,
  marginVertical:5
}
```



6. Ahora procederemos a crear la sección de rutas:

```
01      style={styles.mejores}
02      source={require('./src/img/mejores3.jpg')} />
03    </View>
04  </View>
05
06  <Text style={styles.titulo}>Rutas Turisticas</Text>
07  <View style={styles.listado}>
08    <View style={styles.listaItem}>
09      <Image
10        style={styles.mejores}
11        source={require('./src/img/ruta1.jpg')} />
12    </View>
13
14    <View style={styles.listaItem}>
15      <Image
16        style={styles.mejores}
17        source={require('./src/img/ruta2.jpg')} />
18    </View>
19
20    <View style={styles.listaItem}>
21      <Image
22        style={styles.mejores}
23        source={require('./src/img/ruta3.jpg')} />
24    </View>
25
26    <View style={styles.listaItem}>
27      <Image
28        style={styles.mejores}
29        source={require('./src/img/ruta4.jpg')} />
30    </View>
31  </View>
32 </View>
```

- modificamos las reglas de estilo agregando los siguiente:

```
listaItem:{
  flexBasis:'49%'
},
listado:{
  flexDirection:'row',
  flexWrap:'wrap',
  justifyContent:'space-between'
}
```



7. Vamos agregar un modal para brindar información de las imágenes, para ello debemos realizar lo siguiente en el código:
- agregamos lo siguiente en nuestro código

```
import React,{useState} from 'react';
import {View, StyleSheet,Image, Text, ScrollView, Modal, Button, TouchableHighlight} from 'react-native';

const App = () =>{
  const [modalVisibleplaya, setModalVisibleplaya] = useState(false);
  return(
    <>
    <ScrollView >
      <Modal transparent={true} animationType="slide" visible={modalVisibleplaya} onRequestClose={() => {
        alert('Modal has been closed.')}} >
        <View style={styles.vistaModal}>
          <View style={styles.Modal}>
            <Text style={styles.subtitulo}>Ir a la playa</Text>
            <Text>El Salvador cuenta con hermosas playas a nivel Centroamérica.</Text>
            <Button title="Cerrar" onPress={() =>{setModalVisibleplaya(!modalVisibleplaya)}} ></Button>
          </View>
        </Modal>
      </ScrollView>
    </>
  )
}
```

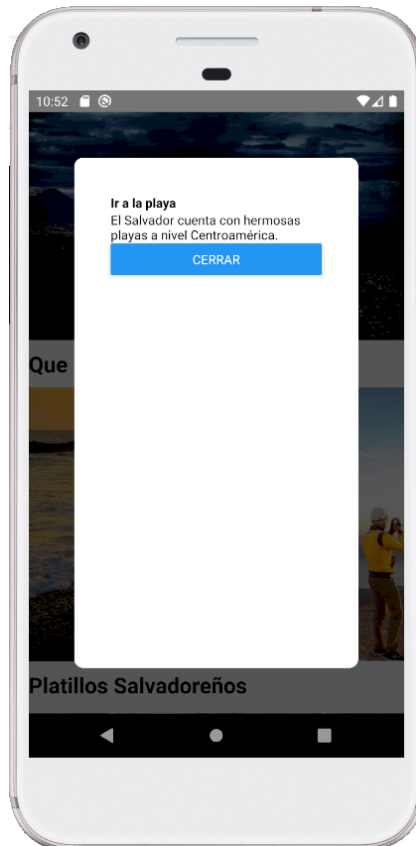
- hacemos cambios en la imagen de actividad 1

```
<View styles={styles.contenedor}>
  <Text style={styles.titulo}>Que hacer en El Salvador</Text>
  <ScrollView horizontal>
    <View>
      <TouchableHighlight
        onPress={() =>{setModalVisibleplaya(!modalVisibleplaya)}}
      >
        <Image
          style={styles.ciudad}
          source={require('./src/img/actividad1.jpg')} />
        </TouchableHighlight>
      </View>
    </ScrollView>
  </View>
```

- agregar los siguientes estilos

```
vistaModal:{
  backgroundColor:'#000000aa',flex:1
},
Modal:{
  backgroundColor:'#fff',margin:50,padding:40,borderRadius:10,flex
:1
},
subtitulo:{
  fontWeight:'bold',
  fontSize:14,
  justifyContent:'center'
},
```

8. deberemos obtener el siguiente resultado, cuando demos clic en la actividad 1:



9. Proceda a agregar modal a 1 elemento de cada sección de la aplicación.

IV. EJERCICIO COMPLEMENTARIO

Realizar una aplicación para un hotel donde muestre el tipo de habitación, los servicios con que cuenta para los huéspedes, muestre lugares de interés cercanos a su hotel.