


Hito 2


Programación de dispositivos móviles

Estudiante: Erick Rodrigo Aranibar Arias
Docente: Ing. William Barra.




Diseñar e implementar el componente CVLogo.

```
import React, {Component} from 'react';import {StyleSheet, View, Text, Image} from
'react-native';import imgLogo from '../Recursos/cochabamba.jpg';
export default class componente extends Component { constructor(props) { super(props); }
render() { return (
  <View style = {styles.container}>
    <Image source = {imgLogo} style = {styles.image}></Image>
    <Text style = {styles.text}>
      Epidemia Corona Virus
    </Text>
  </View>
);
}
}
```




```
const styles = StyleSheet.create(  
  {  
    container: {  
      alignItems: "center",  
      justifyContent: 'center'  
    },  
    image:  
    {  
  
      width: 80,  
      height: 80  
    },  
    text: {  
      color: "black",  
      fontWeight: 'bold',  
      backgroundColor: '#eeec',  
      marginTop: 20  
    }  
  }  
)
```




Diseñar e implementar el componente CVCasos.

```
import React, {Component} from "react";import {StyleSheet, View, Text, TextInput, Image} from
"react-native";import PropTypes from "prop-types";
export default class CVCasos extends Component{
  constructor(props) {  super(props);
    this.Escribir = this.Escribir.bind(this); }
  Escribir (texto) {
    var datos =
    {
      ciudad: this.props.ciudad,
      tipo: this.props.tipo,
      cant: texto
    }
    datos.nombre = [datos.ciudad,datos.tipo].join("_")
    this.props.funcion(datos);
  }
}
```




```
render()
{
  return (
    <View>
      <Text>{this.props.texto}</Text>
      <TextInput
        style = {stylesTextInput.textInput}
        placeholder = {this.props.placeholder}
        secureTextEntry = {this.props.secureTextEntry}
        autoCorrect = {this.props.autoCorrect}
        placeholderTextColor = "#fff"
        underlineColorAndroid = "transparent"
        onChangeText = {this.Escribir}
      ></TextInput>
    </View>
  );
}
```

```
const stylesTextInput = StyleSheet.create({
  textInput: {
    backgroundColor: 'rgba(255, 255, 255, 0.4)',
    alignItems: 'center',
    borderColor: "#999",
    paddingLeft: 40, borderRadius: 15,
    borderBottomWidth: StyleSheet.hairlineWidth,
    marginBottom: 10,
    height: 40
  },
  inlineImg: {
    position: 'absolute',
    zIndex: 99,
    width: 22,
    height: 22,
    left: 10,
    top: 9,
  },
});
CVCasos.propTypes =
{
  source: PropTypes.number.isRequired,
  placeholder: PropTypes.string.isRequired,
  autoCorrect: PropTypes.bool,
  secureTextEntry: PropTypes.bool
}
```



Diseñar e implementar el componente CVCiudad.

```
import React, {Component} from "react";import {StyleSheet, View, Text, TextInput,Image} from "react-native";import CVCasos from "./CVCasos";
export default class CVCiudad extends Component{
  constructor(props) {
    super(props); }
  render()
  {
    return (
      <View>
        <Text>{this.props.nombre}</Text>
        <CVCasos
          funcion = {this.props.funcionConfirmados}
          placeholder = {"0"}
          secureTextEntry = {false}
          autoCorrect = {false}
          texto="Casos Confirmados"
          ciudad = {this.props.nombre}
          tipo = "Confirmados"
        >
      </CVCasos>
    )
  }
}
```



```
<CVCasos
  funcion = {this.props.funcionSospechosos}
  placeholder = {"0"}
  secureTextEntry = {false}
  autoCorrect = {false}
  texto="Casos Sospechosos"
  ciudad = {this.props.nombre}
  tipo = "Sospechosos"
>
</CVCasos>
</View>
);
}
}
```




Diseñar e implementar el componente CVSecreen.

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

```
import CVCasos from "../Componentes/CVCasos"
import CVCiudad from "../Componentes/CVCiudad"
import CVLogo from "../Componentes/CVLogo"
import Boton from "../Componentes/Boton"
```


```
export default class CVPantalla extends Component {
  constructor(props)
  {
    super(props);
    this.state = {
      "Cochabamba_Sospechosos": {},
      "Cochabamba_Confirmados": {},
      "Oruro_Sospechosos": {},
      "Oruro_Confirmados": {},
      "Santa Cruz_Sospechosos": {},
      "Santa Cruz_Confirmados": {}
    }
  }
```

```
    this.CambioCasos = this.CambioCasos.bind(this);
    this.Calcular = this.Calcular.bind(this);
  }
  CambioCasos(datos) {
    const nombre = datos.nombre;
    delete datos.nombre;
    this.setState({[nombre]:datos})
  }
  Calcular() {
    var tipo = this.state["Tipo_De_Búsqueda"];
    tipo = (tipo)? tipo.cant.substr(0,3).toLowerCase():"";
    switch (tipo) {
      case "con":
        tipo = "Confirmados"
        break;
      case "sos":
        tipo = "Sospechosos"
        break;
      default:
        tipo = "Ninguno"
        break;
    }
  }
```

```
if(tipo=="Ninguno") {
  console.log("Ingrese un Tipo de Búsqueda válido")
  ToastAndroid.show("Ingrese un Tipo de Búsqueda
válido", ToastAndroid.LONG);
  return 0;
}
var valores = Object.entries(this.state).map((a)=>
{
  if(a[0]!="Tipo_De_Búsqueda"
  && !(Object.keys(a[1]).length === 0 && a[1].constructor
=== Object) //diferente a objeto vacío.
  && !isNaN(a[1].cant)
  )
  {
    return a[1];
  }
});
valores =
valores.filter(a=>a!=undefined)
.map(a=>{a.cant = parseFloat(a.cant); return a})
.sort((a,b)=>b.cant-a.cant)
.filter(a=>a.tipo===tipo)
.filter(a=>!isNaN(a.cant))
// console.log("valores",valores)
```

```
if(valores.length > 0)
{
  const mensaje = "El valor más alto de tipo "+ tipo +" es: " +
valores[0].cant + " en la ciudad de " + valores[0].ciudad;
  ToastAndroid.show(mensaje, ToastAndroid.LONG);
  console.log(mensaje)
}
else
{
  const mensaje = "ingrese al menos un valor válido del tipo " +
tipo;
  console.log(mensaje);
  ToastAndroid.show(mensaje, ToastAndroid.LONG);
}
render() {
  return (
    <SafeAreaView>
      <ScrollView>
        <View style = {stylesLoginScreen.container}>
          <CVLogo style = {stylesLoginScreen.logo}></CVLogo>
          <View style = {stylesLoginScreen.formCont}>
            <View style = {stylesLoginScreen.form}>
              <Text>
                {this.state.texto}
              </Text>
            </View>
          </View>
        </View>
      </ScrollView>
    </SafeAreaView>
  )
}
```



```

<CVCiudad
  nombre = "Cochabamba"
  funcionConfirmados = {this.CambioCasos}
  funcionSospechosos = {this.CambioCasos}

/>
<CVCiudad
  nombre = "Santa Cruz"
  funcionConfirmados = {this.CambioCasos}
  funcionSospechosos = {this.CambioCasos}
/>
<CVCiudad
  nombre = "Oruro"
  funcionConfirmados = {this.CambioCasos}
  funcionSospechosos = {this.CambioCasos}
/>
<CVCasos
  funcion = {this.CambioCasos}
  placeholder = "Tipo de Búsqueda"
  secureTextEntry = {false}
  autoCorrect = {false}
  texto="Tipo de Búsqueda"
  ciudad = "Tipo"
  tipo = "De_Búsqueda"
/>
<Boton
  titulo = "Calcular"
  _onPress={this.Calcular}
/>

```

```

    </View>
  </View>
</View>
</ScrollView>
</SafeAreaView>
);}
}
const stylesLoginScreen = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: "#469",
    justifyContent: "center",
    alignItems: "center",
    paddingTop: 30
  }, texto:
  {
    fontSize: 45,
    color: "#4f4"
  }, logo: {
  }, form: {
    minWidth: '80%'
  }, formCont: {
    paddingTop: 0,
    flex: 1,
    minWidth: "100%",
    justifyContent: "center",
    alignItems: "center"
  }
});

```