



# Practica 8

Alumno: Valdelamar Martínez Leonardo

Profesor: Ivan Isay Guerra Lopez

Materia: Desarrollo Apps Móviles

Fecha: 16/10/2025

Se explica que TouchableOpacity permite crear botones. Cada botón llama una función que modifica el estado, lo que actualiza el color del texto en pantalla.

También agregamos los botones para el estado de prendido y apagado

```
import { Text, View, StyleSheet, TouchableOpacity, Switch } from 'react-native'
import React, { useState } from 'react'

export default function BotonesScreen() {
  const [prendido, setPrendido] = useState(false);
  const backgroundColor = prendido ? '■#F5F5F5' : '□#573e3eff';
  const textColor = prendido ? '□#000' : '■#FFF';

  return (
    // El signo de interrogación es el operador ternario
    <View style={([styles.container, {backgroundColor}])}>
      <Text style={([styles.texto, {color: textColor}])}>Estado: {prendido ? 'Prendido' : 'Apagado'}</Text>
      <TouchableOpacity
        style={styles.botonEncender}
        onPress={() => setPrendido(true)}>
        <Text style={styles.textoBoton}> Prender </Text>
      </TouchableOpacity>
    </View>
  )
}
```

Agregamos un switch con sus propios estilos para controlar el color del background y las palabras prendido o apagado.

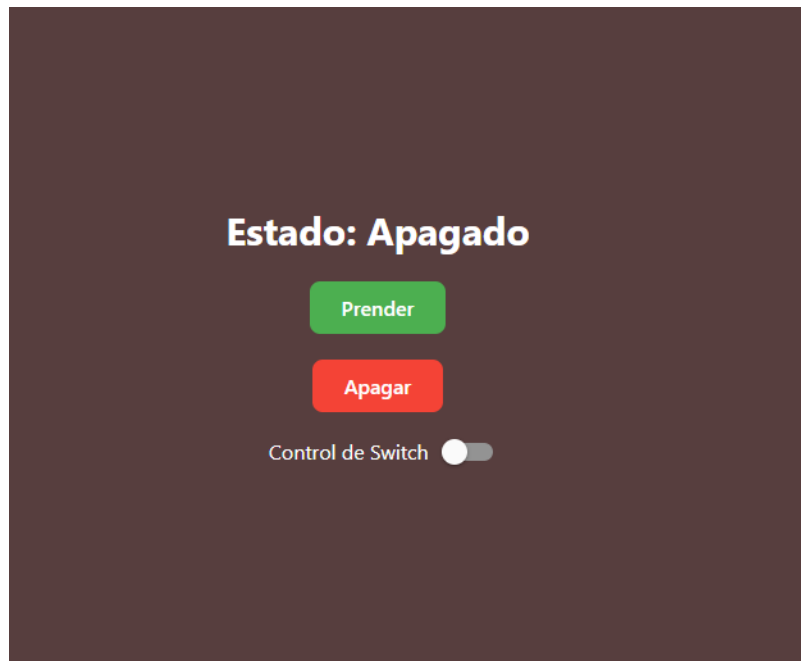
```

    <TouchableOpacity
      style = {styles.botonApagar}
      onPress = (() => setPrendido(false))>
      <Text style={styles.textoBoton}> Apagar </Text>
    </TouchableOpacity>

    <View style={styles.switchContainer}>
      <Text style={([styles.switchLabel, {color: textColor}])}> Control de Switch</Text>
      <Switch value={prendido} onChange={setPrendido} </Switch>
    </View>
  </View>
)
```

En esta parte del código creamos y asignamos las características a cada estilo.

```
}  
const styles = StyleSheet.create({  
  container: {  
    flex: 1,  
    justifyContent: 'center',  
    alignItems: 'center',  
    backgroundColor: '#9bf0ffff',  
  },  
  texto: {  
    fontSize: 30,  
    marginBottom: 20,  
    fontWeight: 'bold',  
  },  
  botonEncender: {  
    backgroundColor: '#4CAF50',  
    paddingVertical: 10,  
    paddingHorizontal: 20,  
    borderRadius: 8,  
    marginBottom: 20,  
  },  
  botonApagar: {  
    backgroundColor: '#F44336',  
    paddingVertical: 10,  
    paddingHorizontal: 20,  
    borderRadius: 8,  
    marginBottom: 20,  
  },  
  textoBoton: {  
    color: 'white',  
    fontSize: 16,  
    fontWeight: 'bold',  
  },  
  switchContainer: {  
    flexDirection: 'row',  
    alignItems: 'center',  
  },  
  switchLabel: {  
    fontSize: 16,  
    marginRight: 10,  
  },  
},
```



Conclusión: Esta práctica nos ayuda a entender de una mejor forma la creación y modificación de los botones en React, aprendimos también a crear switches y a modificarlos de igual forma.