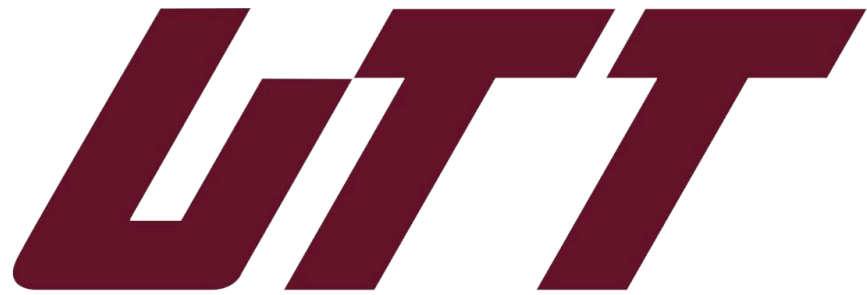


UNIVERSIDAD TECNOLÓGICA DE TIJUANA



Universidad Tecnológica de Tijuana

TSU EN TECNOLOGÍAS DE LA INFORMACIÓN

ÁREA DESARROLLO DE SOFTWARE

MULTIPLATAFORMA

MATERIA:

DISEÑO DE APPS

ESTUDIANTES:

ESTRADA ZARATE DIEGO

DOCENTE:

PARRA GALAVIZ RAY BRUNETT

GRUPO:

4-D BIS

Primer Avance Portafolio

Aplicacion Con un elemento estatico, una imagen.

```
import React from 'react';
import { StatusBar } from 'expo-status-bar';
import { StyleSheet, View, Image } from 'react-native';
import { Button, Text, AppBar, TextInput } from 'react-native-paper';

export default function App() {
  const [text, setText] = React.useState('');
  return (
    <View>
      <AppBar>
        <AppBar.Content title="28 de Enero 2025"/>
      </AppBar>

      <TextInput
        label="Escribe algo"
        value={text}
        onChangeText={text => setText(text)}
        textColor='red'
        style={styles.Input}
      />

      <Text>Mi primera imagen!</Text>
      <StatusBar style="auto" />

      <Button mode onPress={() => alert(`Text: ${text}`)}>
        Show The Text
      </Button>

      <Image
        source={{ uri: 'https://reactnative.dev/img/tiny_logo.png' }}
        style={styles.image}
      />
    </View>
  );
}

const styles = StyleSheet.create({
```

```
export default function App() {
  const [text, setText] = React.useState('');
  style={styles.Input}

  />

  <Text>Mi primera imagen!</Text>
  <StatusBar style="auto" />

  <Button mode onPress={() => alert(`Text: ${text}`)}>
    Show The Text
  </Button>

  <Image
    source={{ uri: 'https://reactnative.dev/img/tiny_logo.png' }}
    style={styles.image}
  />
</View>
);
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#fff',
    alignItems: 'center',
    justifyContent: 'center',
    padding: 16,
  },
  image: {
    width: 100,
    height: 100,
    marginTop: 20,
  },
  Input: {
    marginBottom: 16,
  }
});
```

Explicacion de la aplicacion:

Primero se importan las librerías necesarias de **React** para crear componentes, `expo-status-bar` para manejar la barra de estado, y componentes tanto de `react-native` como de `react-native-paper` para la interfaz.

Se utiliza el hook **useState** para definir el estado `text` y su función actualizadora `setText`. Este estado almacena el texto ingresado por el usuario en el campo de texto **TextInput**.

Todo lo que incluye:

Se usa un componente **View** como contenedor principal, se añade un AppBar de `react-native-paper` para mostrar una barra de encabezado con el título 28 de Enero 2025, se incluye un **TextInput** que permite al usuario ingresar texto. Su valor se vincula al estado `text` y se actualiza cada vez que el usuario escribe.

Se muestra un texto simple indicando "Mi primera imagen!".

Se coloca un `StatusBar` para gestionar la apariencia de la barra de estado.

Se define un botón (`Button`) que, al ser presionado, muestra una alerta con el contenido del estado `text`.

Se renderiza una imagen usando el componente `Image` y se establece su fuente mediante una URL. Se aplican estilos para definir sus dimensiones

Aplicación para mostrar texto

```
import React, {useState} from 'react';// Componente generar y libreria
import { StatusBar } from 'expo-status-bar';
import { StyleSheet, Text, View, TextInput, Button } from 'react-native';

export default function App() {
  const [text,setText] = useState('');
  const [displayText, setDisplayText] = useState('');
  // la funcion no tiene nombre solo los parentecis pero como declaramos como constante la funcion se guarda ahi
  const handlesPress = () => {
    setDisplayText(text);
    setText('');
  }
  return (
    <View style={styles.container}>
      <Text>Hola Mundo</Text>
      <Text>Text to Show: {displayText}</Text>
      <TextInput
        placeholder='Type Something'
        value={text}
        onChangeText={setText}
      />
      <Button title='Click Me!' onPress={handlesPress}/>
      <StatusBar style="auto" />
    </View>
  );
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#4169E1',
    alignItems: 'center',
    justifyContent: 'center',
  },
});
```

Explicacion de la aplicación:

el código es un una actividad que realizamos en clase de un ejemplo básico de una aplicación en React Native que utiliza el hook useState para gestionar dos estados: Uno llamado "text" para almacenar lo que el usuario escribe en un campo de entrada y otro llamado "displayText" para mostrar en pantalla el texto una vez que se presiona un botón.

Primero se importan React y useState, además de componentes esenciales de react-native (como StyleSheet, Text, View, TextInput y Button) y el componente StatusBar desde expo-status-bar para manejar la barra de estado. Dentro del componente principal App se inicializan los estados con useState, asignando una cadena vacía a ambos. Se define una función de flecha, handlesPress, que se ejecuta al presionar el botón y que actualiza displayText con el valor actual de text, limpiando luego el campo de entrada al establecer text en una cadena vacía. En la interfaz, se renderiza un View que usa estilos para centrar sus elementos y darle un fondo azul real.

Todo lo que incluye:

View: Actúa como contenedor para todos los demás elementos, se le aplican estilos para centrar los elementos y asignar un fondo azul real.

Text: Muestra el mensaje "Hola Mundo", otro componente Text muestra el mensaje "Text to Show:" seguido del contenido almacenado en displayText.

TextInput: Permite al usuario ingresar texto. Utiliza el estado text como valor, convirtiéndose en un componente controlado. Cada vez que el usuario escribe, se actualiza el estado text a través de la función onChangeText.

Button: Al presionarse, ejecuta la función handlesPress. Esto transfiere el texto ingresado a displayText y limpia el campo de entrada.

StatusBar: Configura la apariencia de la barra de estado, en este caso con el estilo "auto".

Primera aplicacion

```
import React from 'react';
import { StatusBar } from 'expo-status-bar';
import { StyleSheet, View, Image } from 'react-native';
import { Button, Text, AppBar, TextInput } from 'react-native-paper';

export default function App() {
  const [text, setText] = React.useState('');
  return (
    <View>
      <AppBar>
        <AppBar.Content title="28 de Enero 2025"/>
      </AppBar>

      <TextInput
        label="Escribe algo"
        value={text}
        onChangeText={text => setText(text)}
        textColor="red"
        style={styles.Input}
      />

      <Text>Mi primera imagen!</Text>
      <StatusBar style="auto" />

      <Button mode="text" onPress={() => alert('Text: ' + text)}>
        Show The Text
      </Button>

      <Image
        source={{ uri: 'https://reactnative.dev/img/tiny_logo.png' }}
        style={styles.image}
      />
    </View>
  );
}

const styles = StyleSheet.create({
```

```
  container: {
    flex: 1,
    backgroundColor: 'white',
    alignItems: 'center',
    justifyContent: 'center',
    padding: 16,
  },
  image: {
    width: 100,
    height: 100,
    marginTop: 20,
  },
  Input: {
    marginBottom: 16,
  }
});
```

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

export default function MyTextInput() {
  return (
    <TextInput
      value={text}
      style={styles.input}
    />
  );
}

const styles = StyleSheet.create({
  input: {
    height: 40,
    margin: 17,
    borderRadius: 1,
    padding: 10,
  }
});
```

Esta aplicación tiene dos códigos: el principal y el componente.

Archivo App es lo mismo que con las otras dos aplicaciones se importa el componente `StatusBar` desde `expo-status-bar`, y los componentes `StyleSheet`, `Text`, `View` y `TextInput` desde `react-native`.

Importa el componente `MyTextInput` desde la carpeta "components" (nota: asegúrate de que la ruta sea correcta, ya que "components" podría ser un error tipográfico de "components").

Define el componente `App`, que es el componente principal de la aplicación. Dentro del renderizado del componente `App` se crea un `View` con un estilo definido (container) que centra los elementos. Dentro de este `View` se muestran:
Un `Text` que muestra "Hola mundo".

El componente `MyTextInput`, al que se le pasa una propiedad `text` con el valor "Hola".

Archivo MyTextInput

- Importa los componentes `TextInput` y `StyleSheet` desde `react-native`.
- Define el componente `MyTextInput` como función.
- Dentro del renderizado, devuelve un `TextInput` al que se le asigna el valor de una variable `text` y un estilo definido.
- Se crea un objeto de estilos con `StyleSheet.create` para el `TextInput`, estableciendo propiedades como `altura`, `margen`, `borderRadius` y `padding`.