



PRACTICA 9 – TextInput & Alert

ESTUDIANTE: PÉREZ MELLADO JENNYFER

GRUPO: 211

MAESTRO: IVAN ISAY GUERRA LOPEZ

MATERIA: DESARROLLO DE APPS MÓVILES

FECHA: 19 – 10 – 2025

Preparacion de variables que se usaran:

Empezamos con una constante en la que se marcaran los estados con la variable de nombre, junto con la siguiente constante la cual se nombrara mostrarAlerta y a traves de una función flecha llevara a cabo un proceso de validaciones dadas con un if en el que si no se ha escrito ningun nombre tal como el programa indica, se le mostrara al usuario una alerta la cual indicara este hecho y se le mostrara un mensaje que ademas de marcar dicho error le pedira ingresar un nombre; dentro de esta validación se realiza otra para que la alerta se muestre en los posibles dispositivos, tales como window.alert para que se muestre en el mismo navegador web de windows y Alert.alert para que se muestra y adapte a una pantalla movil.

```
export default function TextInputScreen(){
  const [nombre, setNombre] = useState('');

  const mostrarAlerta = () => {
    if (nombre.trim() === '') {
      if (Platform.OS === 'web') {
        window.alert('ERROR, porfavor ingresa tu nombre');
      } else {
        Alert.alert('ERROR, porfavor ingresa tu nombre')
      }
    }
    else {
      if (Platform.OS === 'web') {
        window.alert(`HOLA ${nombre}`);
      } else {
        Alert.alert(`HOLA ${nombre}`);
      }
    }
  }
}
```

Contenido del return, lo que vera el usuario

En el espacio del return, empezamos con un texto que fungira como titulo, ademas de esto implementamos las etiquetas de TextInput las cuales permitiran al usuario escribir y al programa registrar dicho escrito como la variable nombre, en este se colocan algunas especificaciones como el texto que se mostrar el cual sera “Escribe tu nombre” y que al momento de escribir ya no se vera, value el cual guardara lo que se escriba en la variable antes mencionada, onChangeText que hara uso del estado y un maxLength que nos marcara un maximo de caracteres que podran colocarse.

Despues agregamos un boton el cual mostrara el texto de “Mostrar saludo” y que al ser presionado llamara a la constante antes declarada de mostrarAlerta.

```

return (
  <View>
    <Text>TextScreen</Text>

    <TextInput
      placeholder='Escribe tu nombre:'
      value = {nombre}
      onChangeText={setNombre}
      maxLength={50}
    />

    <Button color='blue' title='Mostrar saludo' onPress={mostrarAlerta}/>
  </View>
)

```

Zona de estilos

En esta zona se van a declarar todos los estilos pertinentes para que el programa sea visualmente atractivo e intuitivo, los nuevos atributos como padding ayudaran a que se marque un espacio definido y por tanto no se vea todo amontonado.

```

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
    backgroundColor: '#33eaf7ff',
    padding: 20,
  },
  titulo: {
    fontSize: 22,
    fontWeight: 'bold',
    marginBottom: 15,
  },
  recuadro: {
    borderWidth: 1,
    borderColor: 'black',
    padding: 10,
    marginBottom: 20,
  },
})

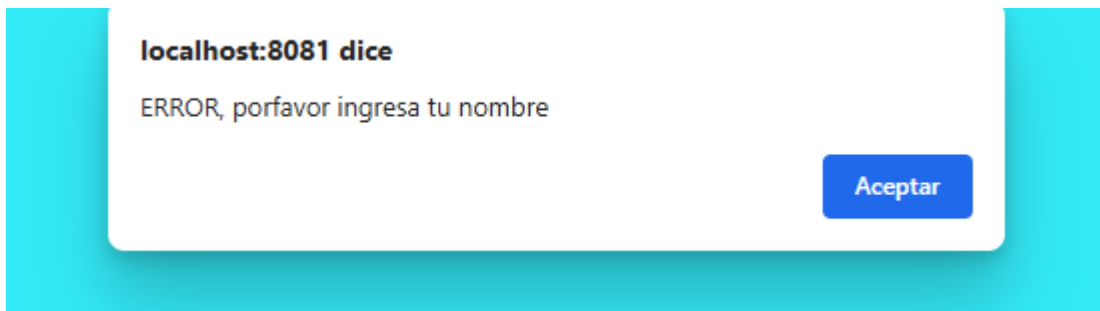
```

En pantalla

De entrada se muestra los mensajes antes dichos y la opción de escribir el nombre



Si no se escribe nada se vera



Por el contrario al colocar el nombre se vera



Segunda parte de la práctica, agregar una función multilinea

Implementamos antes del return, un estado el cual contendrá la variable para que se guarde el texto y como tal el estado, la variable la llamo texto y la implemento dentro del primer if para que le aparezca al usuario un mensaje de error si no ha puesto nada en cualquiera de los dos campos, y en el mensaje cuando ambos están se implementa el texto en el mensaje de alerta que se mostrara.

```
export default function TextInputScreen(){
  const [nombre, setNombre] = useState('');

  const [texto, setTexto] = React.useState('');

  const mostrarAlerta = () => {
    if ((nombre.trim()=== '') || (texto.trim()=== '')) {
      if(Platform.OS === 'web') {
        window.alert ('ERROR, porfavor ingresa tu nombre y un texto');
      } else {
        Alert.alert ('ERROR, porfavor ingresa tu nombre y un texto')
      }
    }
    else {
      if (Platform.OS === 'web') {
        window.alert(`Hola ${nombre} tu texto es ${texto}`);
      } else {
        Alert.alert(`Hola ${nombre} tu texto es ${texto}`);
      }
    }
  }
}
```

Más tarde en el return se lleva a cabo la implementación de dicho input, solo modificando y agregando un par de atributos que antes no lo hicimos, estos son, multiline para registrar que se trata de un texto multilinea, es decir más extenso, numberOfLines el cual marcara las líneas permitidas para el texto, solo una modificación en los caracteres que acepta para tener un texto más extenso.

```
<TextInput
  style={styles.recuadro}
  placeholder='Escribe un texto:'
  multiline
  numberOfLines={4}
  maxLength={100}
  value={texto}
  onChangeText={setTexto}
/>
```

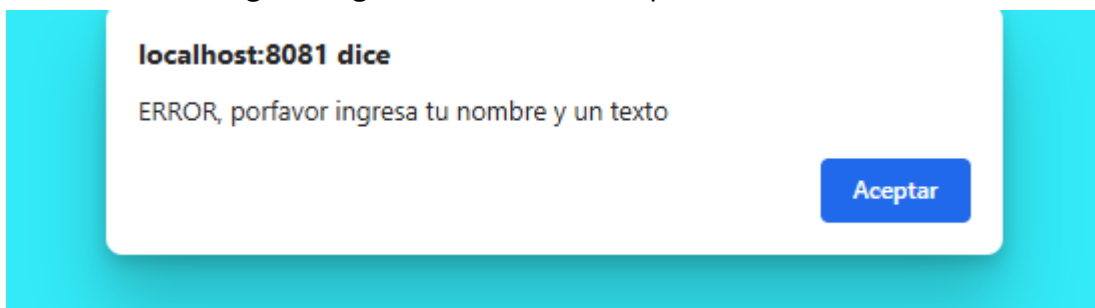
Visualización en pantalla

De forma general



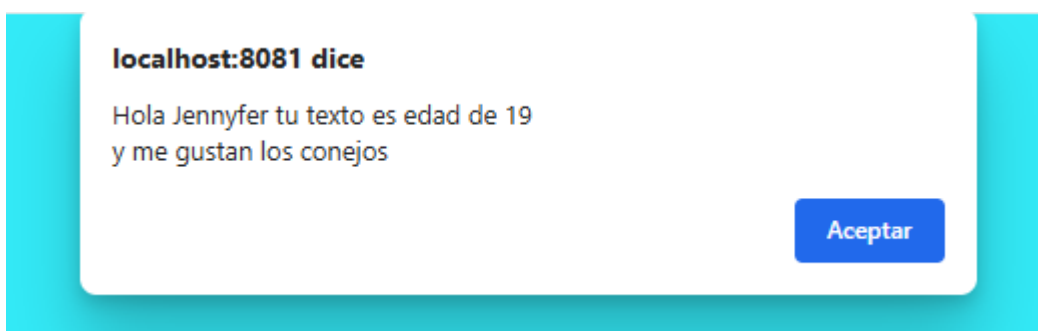
The screenshot shows a web application titled "TextScreen" on a light blue background. It features two input fields: a smaller one labeled "Escribe tu nombre:" and a larger one labeled "Escribe un texto:". Below these fields is a red button with the text "MOSTRAR SALUDO" in white capital letters.

Si no se coloca algo en algunos de los dos campos



This screenshot shows the application's response to missing input. A white message box with a red border and a red shadow contains the text "localhost:8081 dice" in bold, followed by "ERROR, porfavor ingresa tu nombre y un texto" in a smaller font. A red "Aceptar" button is located at the bottom right of the message box.

Si se colocan ambos



This screenshot shows the application's response when both fields are filled. The white message box contains the text "localhost:8081 dice" in bold, followed by "Hola Jennyfer tu texto es edad de 19" and "y me gustan los conejos" on separate lines. A red "Aceptar" button is at the bottom right.

Tercera parte de la práctica

Para la siguiente parte, agregamos un nuevo `textInput` el cual al escribir en el teléfono celular solo nos dará acceso al teclado numérico, para ello hacemos todo el mismo proceso anterior, agregando una nueva variable que guarde dicha información para más tarde mostrarla a modo de alerta en la pantalla, por lo que queda de esta forma.

```
export default function TextInputScreen(){
  const [nombre, setNombre] = useState('');

  const [texto, setTexto] = useState('');

  const [number, setNumber] = useState('');

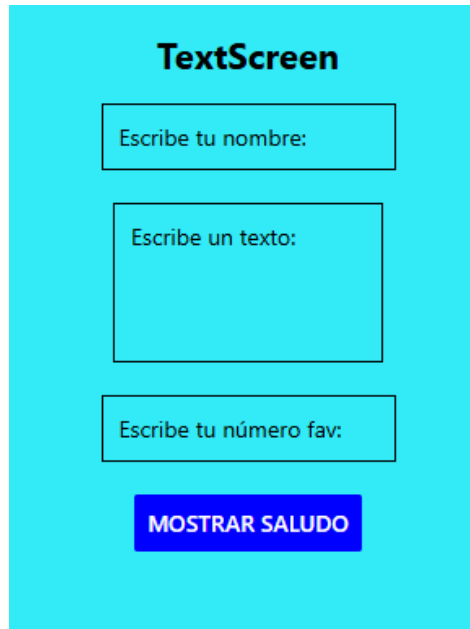
  const mostrarAlerta = () => {
    if ((nombre.trim()=== '') || (texto.trim()=== '') || (number.trim()=== '')) {
      if(Platform.OS === 'web') {
        window.alert ('ERROR, porfavor ingresa tu nombre, un texto y un número');
      } else {
        Alert.alert ('ERROR, porfavor ingresa tu nombre, un texto y un número')
      }
    }
    else {
      if (Platform.OS === 'web') {
        window.alert(`Hola ${nombre} tu texto es ${texto} y tu número fav es ${number}`);
      } else {
        Alert.alert(`Hola ${nombre} tu texto es ${texto} y tu número fav es ${number}`);
      }
    }
  }
}
```

De esta misma forma se agrega una nueva etiqueta de `textInput`, en la cual se le agregara un nuevo atributo que controlara el teclado tal como se menciono anteriormente, dicho atributo es `inputMode` en el que se especifica que parte del teclado quiere, en este caso ponemos `numeric` lo que habilita solo un teclado numérico.

```
<TextInput
  style={styles.recuadro}
  placeholder='Escribe tu número fav:'
  value = {number}
  onChangeText={setNumber}
  maxLength={50}
  inputMode='numeric'
/>
```

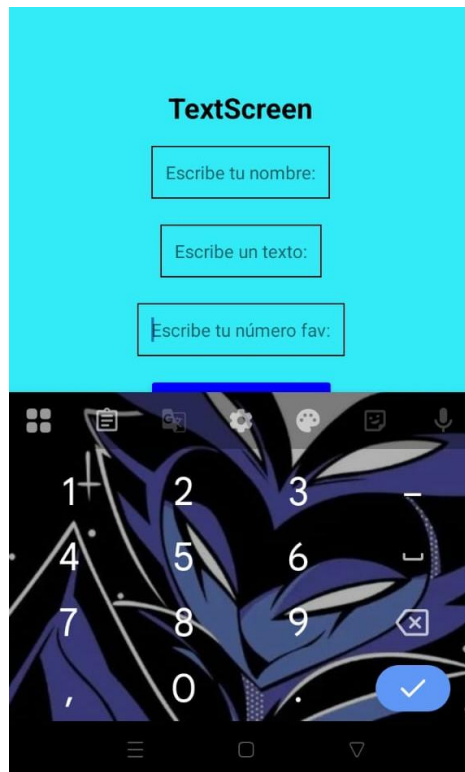
Visualización en pantalla

Para la visualización podemos encontrar la nueva opción de escritura de un número



The image shows a desktop monitor displaying the 'TextScreen' application. The app has a light blue background and the title 'TextScreen' at the top. It contains three input fields: 'Escribe tu nombre:', 'Escribe un texto:', and 'Escribe tu número fav:'. Below these fields is a red button labeled 'MOSTRAR SALUDO'.

Y al usarlo en teléfono veremos como nos da un teclado numérico para escribir en dicho campo.



Ultima parte de la práctica

Como ultima parte implementamos un ultimo atributo a nuestro texInput el cual mediante un atributo permite cifrar lo que se esta escribiendo, esto logrando el efecto el cual al escribir una contraseña en lugar de mostrar dichas letras nuestro solo puntos negros, para ello al igual que en los anteriores puntos empezamos agregando una constante que definirá a dicha variable y su estado.

```
export default function TextInputScreen(){
  const [nombre, setNombre] = useState('');

  const [texto, setTexto] = useState('');

  const [number, setNumber] = useState('');

  const [clave, setClave] = useState('');

  const mostrarAlerta = () => {
    if ((nombre.trim()=== '') || (texto.trim()=== '') || (number.trim()=== '') || (clave.trim()=== '')) {
      if(Platform.OS === 'web') {
        window.alert('ERROR, porfavor ingresa tu nombre, un texto, un número y una clave');
      } else {
        Alert.alert('ERROR, porfavor ingresa tu nombre, un texto, un número y una clave')
      }
    } else {
      if (Platform.OS === 'web') {
        window.alert(`Hola ${nombre}. Tu texto es: ${texto}. Tu número fav es ${number}. Y tu clave es: ${clave}`);
      } else {
        Alert.alert(`Hola ${nombre}. Tu texto es: ${texto}. Tu número fav es ${number}. Y tu clave es: ${clave}`);
      }
    }
  }
}
```

Más adelante se agrega una nueva etiqueta de textInput en la cual se implementará un nuevo atributo que es **secureTextEntry** lo que como se menciono antes realizara la conversión del texto, este atributo tiene un valor predeterminado de false para que no se ejecute por lo que se realizara su asignación a true.

```
<TextInput
  style={styles.recuadro}
  placeholder='Escribe tu clave:'
  value = {clave}
  onChangeText={setClave}
  maxLength={50}
  secureTextEntry={true}
/>
```

Visualización en pantalla

Con los cambios implementados en la pantalla aparecerá el nuevo campo de clave para escribir en el mismo y al hacerlo veremos solo unos puntitos negros, de forma inicial también nos aparecerá el icono de un ojo, el cual al seleccionarlo mostrará o volverá a cifrar la contraseña.

TextScreen

Este es un texto
Que contiene
Unas 4 líneas
Y muchos caracteres

.....

MOSTRAR SALUDO

TextScreen

Este es un texto
De tipo multilinea
Ósea tiene varias líneas
Y muchos caracteres

ElefanteMorado8

👁

MOSTRAR SALUDO

Finalmente, con todos los elementos completados, el mensaje de alerta se vera de la siguiente forma.

localhost:8081 dice
Hola Jennyfer. Tu texto es: Este es un texto
De tipo multilinea
Ósea tiene varias líneas
Y muchos caracteres. Tu número fav es 8. Y tu clave es: ElefanteMorado8

Aceptar

Conclusión: En esta práctica realmente aprendí mucho, además de la forma para ingresar texto, también las múltiples propiedades y atributos que esta misma puede tener y que permite que se logren cosas distintas, como el teclado numérico.