
UNIVERSIDAD TECNOLÓGICA DE TIJUANA

TSU SOFTWARE

APP DESIGN

PREPARED BY:

LEMUS TARANGO NAOMI DENISSE

IN THE SUBJECT OF:

EVIDENCE PORTFOLIO

GIVEN BY THE TEACHER:

RAY BRUNETT PARRA GALAVIZ

GROUP:

4 B

Tijuana, Baja California, FEBRUARY 4, 2025

Content

I. First exercise.....	3
Figure 1. folders.....	3
Figure 2. Code of the first act.....	4
Figure 3. Ejection	5
II. Second exercise.....	6
Figure 4. folders.....	6
Figure 5. Code of the image.....	7
Figure 6. Ejection	8
III. third exercise.....	9
Figure 7. folders.....	9
Figure 8. Code.....	10
Figure 8. Code.....	11
Figure 9. ejecution.....	12
IV. quarter exercise.....	13
Figure 10. folders.....	14
Figure 11. Code of the quarter.....	14
Figure 12. Ejection	15

I. First exercise

In the first exercise, operations such as attaching buttons and texts to a box were performed, which was to display a TextInput in which the text would be shown. The button was used as a simulation in which it was shown on a cell phone model.

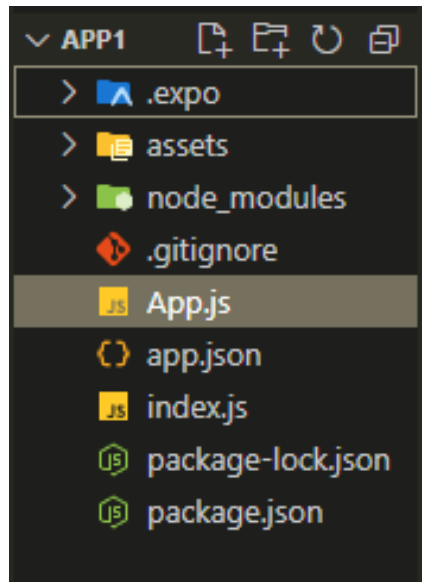


Figure 1. folders.

```

Appjs > App > handlePress
1  import React, {useState} from 'react';
2  import { StatusBar } from 'expo-status-bar';
3  import { StyleSheet, Text, TextInput, Button, View } from 'react-native';
4
5  export default function App() {
6
7      const [text, setText] = useState('');
8      const [displayText, setDisplayText] = useState('');
9
10     const handlePress = () => {
11         setDisplayText(text);
12         setText('');
13     }
14     return (
15         <View style={styles.container}>
16             <TextInput
17                 style={styles.input}
18                 placeholder="Type Something"
19                 value={text}
20                 onChangeText={setText}
21             />
22             <Button title="Click Me!" onPress={handlePress}/>
23             <Text style={styles.resultText}>Text to Show: {displayText}</Text>
24             <StatusBar style="auto" />
25         </View>
26     );
27 }
28
29 const styles = StyleSheet.create({
30     container: {
31         flex: 1,
32         backgroundColor: '#fff',
33         alignItems: 'center',
34         justifyContent: 'center',
35         padding: 16,
36     },
37     input: {
38         width: '80%',
39         height: 40,
40         borderWidth: 1,
41         borderColor: 'gray',
42         marginBottom: 16,
43         marginLeft: 8,
44     },
45     resultText: {
46         marginTop: 16,
47         fontSize: 16,
48     },
49 },

```

Figure 2. Code of the first act.

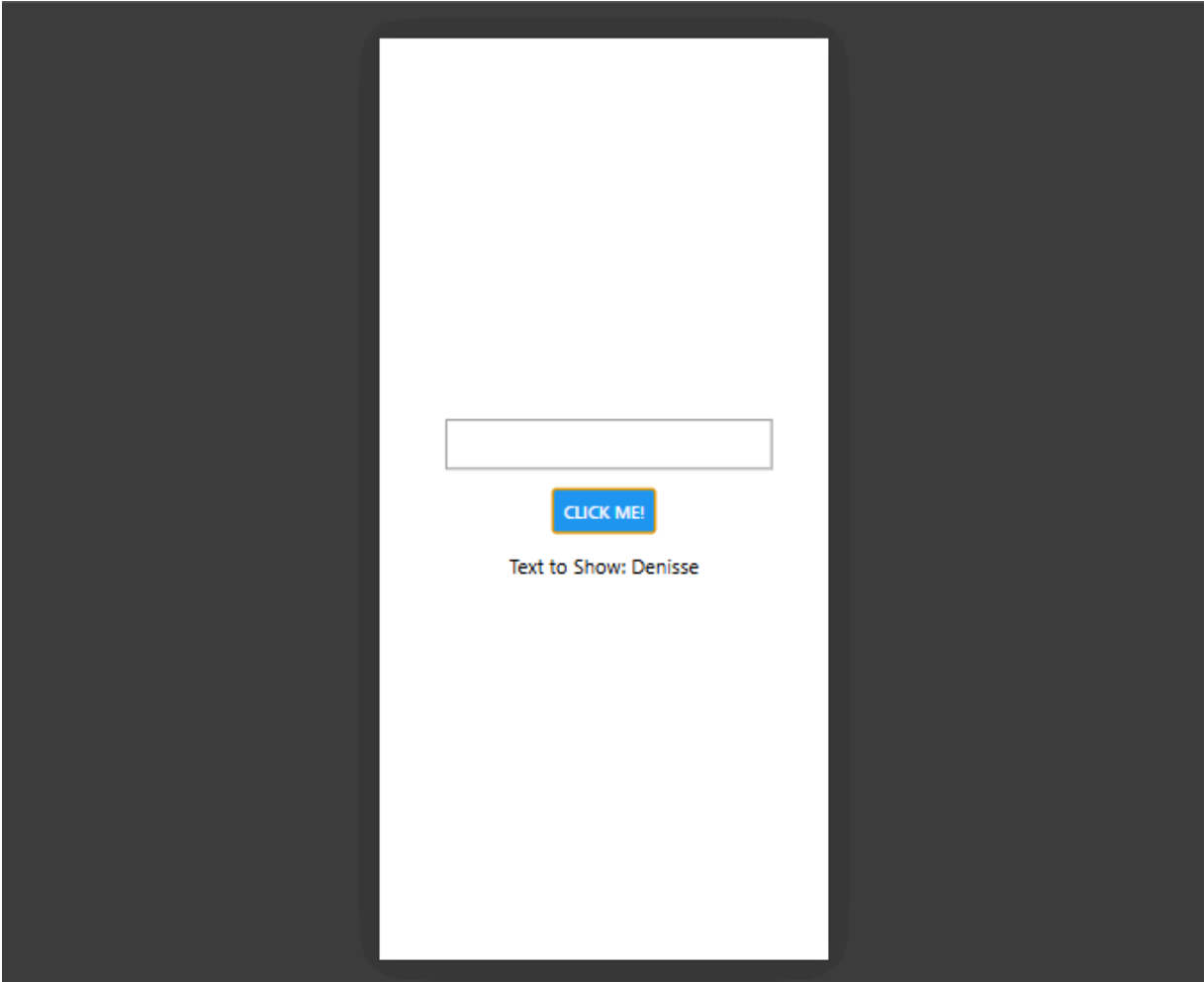


Figure 3. Ejection .

II. Second exercise

In this exercise, the action of attaching images was performed by downloading them as images on the laptop and by using links on a website. The following commands were used: `<image source={require('./assets/connor.jpg')} />` for downloaded images and `const third_image = "https://i.pinimg.com/originals/65/85/70/6585701dd930320bad656b80b4f8cdb9.jpg"` for image URLs.

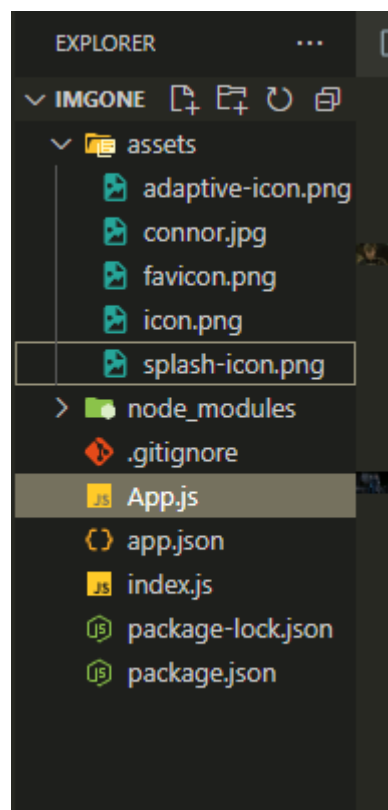


Figure 4. folders.

```

App.js > App
1  import { StatusBar } from 'expo-status-bar';
2  import { StyleSheet, Text, View } from 'react-native';
3
4  const third_image = "https://i.pinimg.com/originals/65/85/70/6585701dd930320bad656b80b4f8cdb9.jpg"
5
6  export default function App() {
7    return (
8      <View style={styles.container}>
9        <Text>Open up App.js to start working on your app!</Text>
10       <image source={require('./assets/connor.jpg')} />
11       <Text style={style.titttle}>Source: Thrid image!</Text>
12       <image source ={{uri: third_image}} />
13       <StatusBar style="auto" />
14     </View>
15   );
16 }
17
18 const styles = StyleSheet.create({
19   container: {
20     flex: 1,
21     backgroundColor: '#fff',
22     alignItems: 'center',
23     justifyContent: 'center',
24   },
25
26   title: {
27     fontSize: 24,
28     fontWeight: 'bold',
29     marginBottom: 20,
30   },
31 },
32   image: {
33     width: 200,
34     height: 250,
35   },
36 },
37 });
38

```

Figure 5. Code of the image.

connor



Leon Kennedy



Figure 6. Ejection .

III. third exercise

In this practice, an exercise was carried out which consists of making a folder called components which would have a file called MyForm.js, this file has a form which would be displayed by making an extension from the app.js file to MyForm.js.

```
import MyForm from './components/MyForm';
```

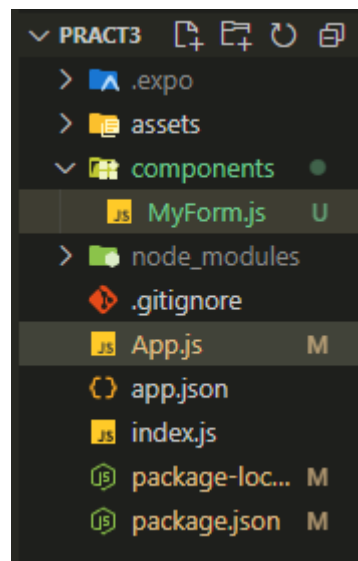


Figure 7. folders.

```

App.js > ...
import React, {useState} from 'react';
import { StatusBar } from 'expo-status-bar';
import { StyleSheet, View } from 'react-native';
import MyForm from './components/MyForm';

export default function App() {
  return (
    <View style={styles.container}>
      <MyForm/>
      <StatusBar style="auto" />
    </View>
  );
}

const styles = StyleSheet.create({
  container: {
    backgroundColor: 'fff',
  },
});

```

Figure 8. Code.

```

components > MyForm.js > styles > resultText
1  import React, {useState} from 'react';
2  import {StyleSheet, Text,TextInput,Button,View} from 'react-native';
3
4  export default function MyForm(){
5
6      const [text, setText] = useState('');
7      const [displayText, setDisplayText] = useState('');
8
9      const handlePress = () => {
10         setDisplayText(text);
11         setText('');
12     }
13
14     return(
15         <View style={styles.container}>
16             <TextInput
17                 style={styles.input}
18                 placeholder='Type Something'
19                 value={text}
20                 onChangeText={setText}
21             />
22             <Button title="Click me!" onPress={handlePress} />
23             <Text>Text to show: {displayText}</Text>
24         </View>
25     );
26 }
27
28 const styles = StyleSheet.create({
29     container: {
30         flex: 1,
31         alignItems: 'center',
32         justifyContent: 'center',
33         padding: 16,
34     },
35     input:{
36         width:'80%',
37         height: 40,
38         borderWidth:1,
39         borderColor:'gray',
40         marginBottom:16,
41         paddingLeft:8,
42     },
43     resultText:
44         marginTop:16,
45         fontSize:16,
46     },
47 });

```

Figure 8. Code.

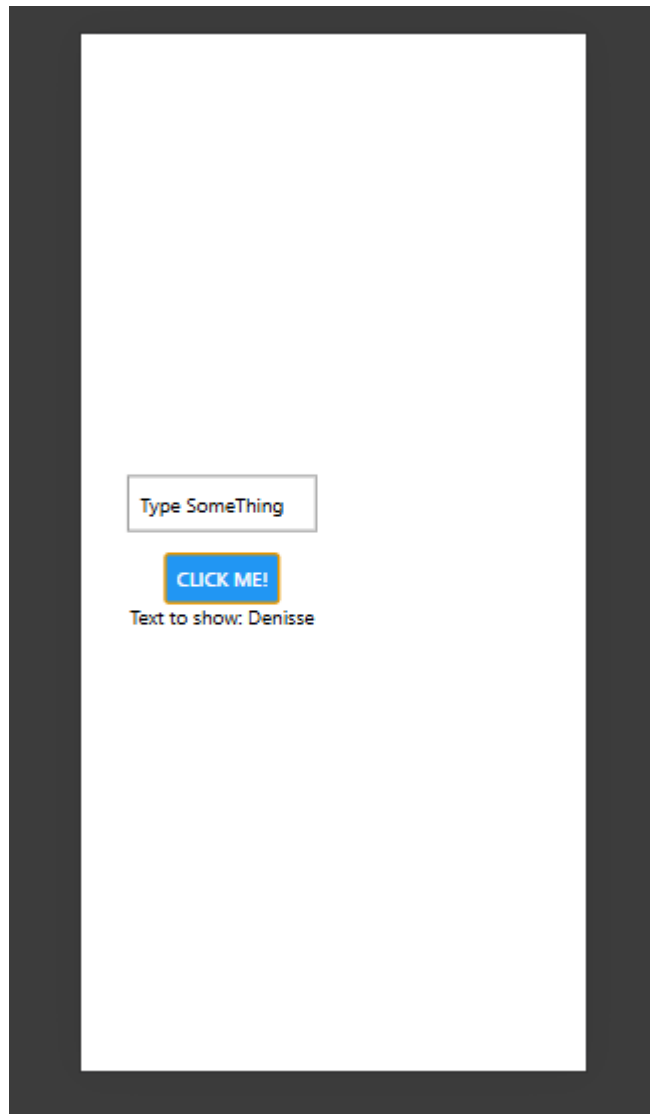


Figure 9. ejecution.

IV. quarter exercise

In this exercise, the action of adding a text input and placing an alert was carried out, which is designed to give a warning, different codes are used such as:

```
const [text, setText] = React.useState('');
```

which is used to store the text that is written.

```
<TextInput  
  style={styles.input}  
  label='Type something'  
  value={text}  
  textColor="blue"  
  onChangeText={text => setText(text)}  
/>
```

It is used to give style to the text.

```
<Button mode='contained' onPress={() => alert(`Text: ${text}`)}>  
  Type something  
</Button>
```

to be able to view the alert.

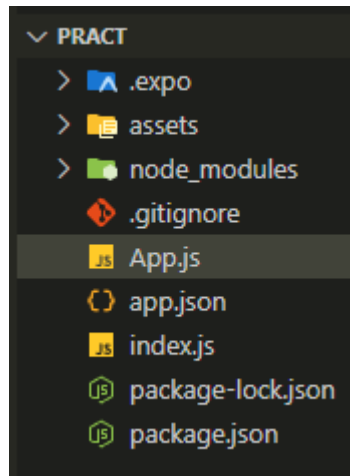


Figure 10. folders.

```
Appjs x
Appjs > App
1  import React from 'react';
2  import { StatusBar } from 'expo-status-bar';
3  import { StyleSheet, View } from 'react-native';
4  import { Button, TextInput, AppBar } from 'react-native-paper';
5
6
7  export default function App() {
8
9    const [text, setText] = React.useState('');
10
11    return (
12      <View style={styles.container}>
13        <AppBar>
14          <AppBar.Content title="Just real" />
15        </AppBar>
16        <TextInput
17          style={styles.input}
18          label="Type something"
19          value={text}
20          textColor="blue"
21          onChangeText={text => setText(text)}
22        />
23      <Button mode="contained" onPress={() => alert("Text: ${text}")}>
24        Type something
25      </Button>
26      <StatusBar style="auto" />
27    </View>
28  );
29 }
30
31
32 const styles = StyleSheet.create({
33   container: {
34     flex: 1,
35     backgroundColor: '#fff',
36     padding: 16,
37     justifyContent: 'center',
38   },
39   input: {
40     marginBottom: 16,
41     width: '100%',
42   }
43 });
44
```

Figure 11. Code of the quarter.

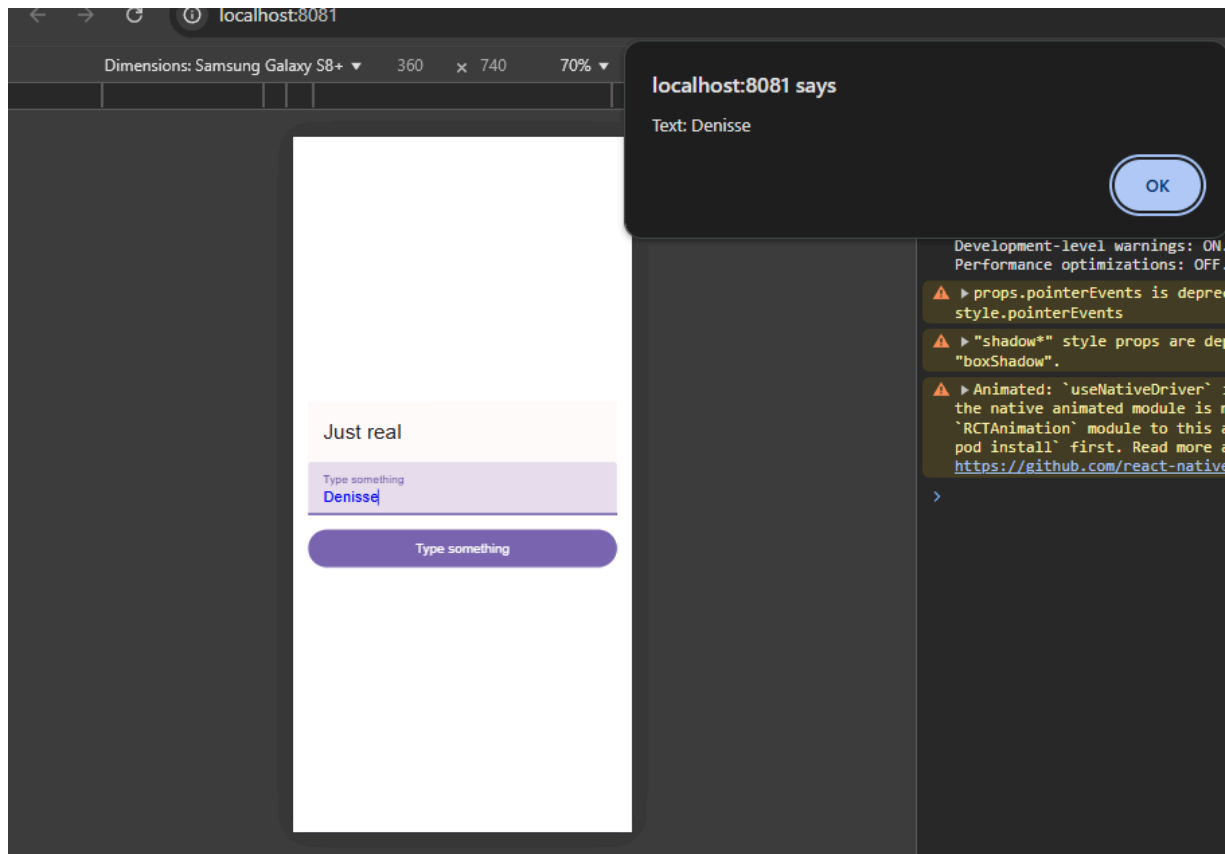


Figure 12. Ejection .