

Reporte Mensual de actividades No. 2:

Marthon Leobardo Yañez Martínez

Resumen

Durante este mes desarrollé en conjunto las pantallas que se mostrarán a los usuarios en la vista forntal, indexadas en GitHub como: HomeScreen y LoginScreen

Código

LoginScreen

Dentro del archivo LoginScreen.js se migró el código de App.js del sprint anterior, con algunas modificaciones que se explicarán a continuación:

Lo primero fueron las importaciones:

```
import React, { useState } from 'react';
import {
  Text,
  View,
  StyleSheet,
  Image,
  TextInput,
  Pressable,
} from 'react-native';
```

Posteriormente la redeclaración de la función para ser llamada desde otro bloque de código:

```
export default function LoginScreen({ onLogin }) {  
  const [nombre, setNombre] = useState('');  
  const [edad, setEdad] = useState('');  
  
  const handlePress = () => {  
    onLogin(nombre, edad);  
  };  
}
```

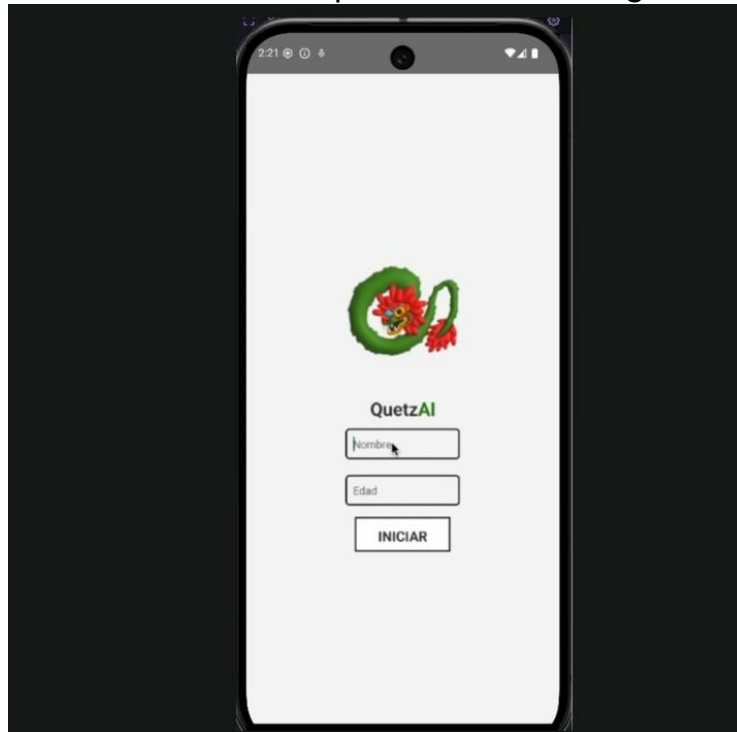
A continuación, el código que representa la primera pantalla para el usuario:

```
return (  
  <View style={styles.container}>  
    <Image style={styles.logo} source={require(  
      '../img/QuetzAI.png')} />  
    <Text style={styles.textLogo1}>  
      Quetz  
    <Text style={styles.textLogo2}>AI</Text>  
    </Text>  
    <TextInput  
      style={styles.input}  
      placeholder="Nombre"  
      keyboardType="default"  
      onChangeText={setNombre}  
      value={nombre}  
    />  
    <TextInput  
      style={styles.input}  
      placeholder="Edad"  
      keyboardType="numeric"  
      onChangeText={setEdad}  
      value={edad}  
    />  
    <Pressable  
      style={({pressed}) => [  
        {  
          backgroundColor: pressed ? 'black' : '#fff',  
          color: pressed ? '#fff' : 'black',  
        },  
        styles.button,  
      ]}  
      onPress={handlePress}>  
      <Text style={styles.textInPlace}>INICIAR</Text>  
    </Pressable>  
  </View>  
)  
);  
}
```

Declaración de los estilos:

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
    backgroundColor: '#f5f5f5',
  },
  textLogo1: {
    fontSize: 24,
    color: '#333',
    fontWeight: 'bold',
  },
  textLogo2: {
    fontSize: 24,
    color: '#008000',
    fontWeight: 'bold',
  },
  textInPlace: {
    fontSize: 18,
    color: '#333',
    fontWeight: 'bold',
  },
  logo: {
    width: 200,
    height: 200,
    alignContent: 'center',
  },
  input: {
    height: 40,
    width: 150,
    margin: 10,
    borderWidth: 2,
    padding: 10,
    borderRadius: 5,
  },
  button: {
    padding: 10,
    borderRadius: 0,
    borderWidth: 2,
    fontSize: 18,
    height: 45,
    width: 125,
    justifyContent: 'center',
    alignItems: 'center',
    margin: 5,
  },
});
```

Finalmente, este fue el resultado de la aplicación de este código.



HomeScreen

En este código se creó para que el usuario pueda escanear los NFC y cierra la sesión:

En este bloque se muestran las importaciones:

```
import React from 'react';
import {
  Text,
  View,
  StyleSheet,
  Image,
  Pressable,
} from 'react-native';
```

Posteriormente en esta imagen se muestra el código implementado, así como la declaración de la función para que pueda ser invocada en cualquier momento en otros bloques del código:

```
export default function HomeScreen({
  nombre,
  edad,
  onNavigateToNfc,
  onLogout
}) {
  return (
    <View style={styles.container}>
      <Image style={styles.logo} source={require(
        './img/QuetzAI.png')} />
      <Text style={styles.textLogo1}>
        ¡Bienvenido, <Text style={styles.textLogo2}>{nombre}
      </Text>
      </Text>
      <Text style={styles.subtitle}>Edad: {edad} años</Text>

      <Pressable
        style={({pressed}) => [
          {
            backgroundColor: pressed ? '#008000' : '#fff',
            borderColor: '#008000',
          },
          styles.button,
        ]
        onPress={onNavigateToNfc}>
        <Text style={styles.textInPlace, {color: '#008000'}}
      >>Escanear NFC</Text>
      </Pressable>

      <Pressable
        style={({pressed}) => [
          {
            backgroundColor: pressed ? '#9b59b6' : '#fff',
            borderColor: '#9b59b6',
          },
          styles.button,
        ]
        onPress={() => onNavigateToNfc(true)}>
        <Text style={styles.textInPlace, {color: '#9b59b6'}}
      >> Modo Demo</Text>
      </Pressable>

      <Pressable
        style={({pressed}) => [
          {
            backgroundColor: pressed ? '#ff4444' : '#fff',
            borderColor: '#ff4444',
          },
          styles.button,
        ]
        onPress={onLogout}>
        <Text style={styles.textInPlace, {color: '#ff4444'}}
      >>Cerrar Sesión</Text>
      </Pressable>
    </View>
  );
}
```

A continuación, se muestran los estilos que le dan visibilidad a la interfaz de modo que sea amigable e intuitiva:

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
    backgroundColor: '#f5f5f5',
  },
  textLogo1: {
    fontSize: 24,
    color: '#333',
    fontWeight: 'bold',
  },
  textLogo2: {
    fontSize: 24,
    color: '#008000',
    fontWeight: 'bold',
  },
  subtitle: {
    fontSize: 16,
    color: '#666',
    marginBottom: 10,
    textAlign: 'center',
  },
  textInPlace: {
    fontSize: 18,
    color: '#333',
    fontWeight: 'bold',
  },
  logo: {
    width: 200,
    height: 200,
    alignContent: 'center',
  },
  button: {
    padding: 10,
    borderRadius: 0,
    borderWidth: 2,
    fontSize: 18,
    height: 45,
    width: 125,
    justifyContent: 'center',
    alignItems: 'center',
    margin: 5,
  },
});
```

Finalmente, este fue el resultado de la integración del código, donde se muestra un saludo personalizado, se reitera su edad, confirmando de una manera visual que se ha guardado exitosamente la información.

