



Evaluacion Hito 3

Estudiante: Maria Bernarda Serrano Larrea

Docente: Ing. William Barra

Materia: PDM

Para iniciar el examen

Para poder realizar el examen se instalaron y configuraron 2 paquetes de npm: React Navigatos y Firebase .

Con estos dos instalados se procedio con la evaluacion.





Componente Button

Desarrollo

Para este componente solamente implemente un Touchable opacity de react native y Text para poder realizar un botón, los props que recibirá esta clase son: un Texto y la acción del evento onPress.

```
import React, { Component } from 'react';
import PropTypes from 'prop-types';
import { StyleSheet, View, TouchableOpacity, Text } from 'react-native';
import Colors from '../config/colors';

class button extends Component {
  constructor(props) {
    super(props);
  }

  render() {
    return (
      <View>
        <TouchableOpacity style={stylesButton.container} onPress={this.props.onPress}>
          <Text style={stylesButton.text}>{this.props.titleButton}</Text>
        </TouchableOpacity>
      </View>);
  }
}
```

Imports de librerías y componentes

Clase button

```
const stylesButton = StyleSheet.create({
  container: {
    width: '100%',
    alignItems: 'center',
    paddingVertical: 20,
    borderRadius: 10,
    borderWidth: StyleSheet.hairlineWidth,
    borderColor: Colors.white,
    paddingLeft: 30,
    paddingRight: 30,
  },
  text: { color: Colors.white, textAlign: 'center', height: 20 },
});

button.propTypes = {};
```

Estilos para el boton

**El Diseño debería
quedar algo así:**





Componente Description.js

Desarrollo

Este componente necesita 3 cosas contener el texto principal, una imagen y un texto de descripción, para esto se realizo una clase que recibió las propiedades de la vista principal.

```
import React from 'react';
import {
  View,
  Text,
  TouchableOpacity,
  Dimensions,
  StyleSheet,
  Image
} from 'react-native';
//imagenes
import Img from '../assets/img2.png'
import Colors from '../config/colors'
import Constants from '../config/constants'
```

Imports de
librerías y
componentes

```
function DefensaHito3({ img, text, text1, text2 }) {
  return (
    <View style={styles.container}>
      <View style={styles.header}>
        <Image
          source={img}
          style={styles.image}
        />
      </View>
      <View style={styles.footer}>
        <Text style={styles.title}>{text}</Text>
        <Text style={styles.text}>{text1}</Text>
        <Text style={styles.text}>{text2}</Text>
      </View>
    </View>
  );
};
```

Funcion para la
descripción, requiere
los props img, text,
text1, text2

```
const { height } = Dimensions.get("screen");
const height_logo = height * 0.28;
const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: Colors.Naranja,
  },
  header: {
    flex: 1,
    paddingLeft: '50%'
  },
  footer: {
    flex: 1,
    paddingTop: '100%'
  },
  logo: {
    width: height_logo,
    height: height_logo
  },
  image: {
    flex: 0,
    //ajustar el contenido de la imagen
    resizeMode: 'stretch'
  },

```

Estilos para el boton

```
title: {
  alignSelf: 'center',color: Colors.white,
  fontSize: 30,
  fontWeight: 'bold'
},
text: {
  alignSelf: 'center',
  textAlign: 'center',
  color: Colors.white,
  fontSize: 18,
  marginTop: 5
},
button: {
  alignItems: 'flex-end',
  marginTop: 100,
  flex: 1,
  flexDirection: 'row',
  justifyContent: 'space-between',
},
signIn: {
  width: '20%',
  alignItems: 'center',
  marginBottom: 12,
  paddingVertical: 12,
  borderRadius: 10,
  borderWidth: StyleSheet.hairlineWidth,
  borderColor: Colors.white,
},
textSign: {
  color: Colors.white, textAlign: 'center', height: 20
}
});
```

El Diseño debería quedar algo así:





Componente ButtonsFooter.js

Desarrollo

Este requiere el componente button y será el componente que se visualice al pie de la aplicación para hacer uso de la navegacion entre vistas. Este enviara la prop de texto al componente button con el texto obtenido desde constants.js sin embargo el evento onPress aun no se ha enviado.

```
import React, {Component} from 'react';
import PropTypes from 'prop-types';
import {StyleSheet, View, TouchableOpacity, Text} from 'react-native';
import Colors from '../config/colors';
import Button from './Button';
import constants from '../config/constants';
```

Imports de
librerías y
componentes

Clase para los
botones del footer

```
class button extends Component {
  constructor(props) {
    super(props);
  }
  render() {
    return (
      <View style={styles.button}>
        <Button
          titleButton={constants.STRING.S.PREV}
          onPress={this.props.onPress}
        />
        <Button
          titleButton={constants.STRING.S.NEXT}
          onPress={this.props.onPress2}
        />
      </View> );}}

```

```
const styles = StyleSheet.create({
  button: {
    marginTop: 100,
    flex: 1,
    flexDirection: 'row',
    justifyContent: 'space-between',
  },
  container: {
    width: '50%',
    alignItems: 'center',
  },
});

button.propTypes = {};
```

Estilos para el botón footer, el flexdirection hizo que se acomodaran y el justifycontent hizo que tengan esa distancia.

El Diseño debería quedar algo así:





Componente OnboardingFactory.js

Desarrollo

Este requiere es la combinación de los anteriores componentes, por lo que solo será necesario importarlos y los props seguirán en espera.

```
import React from 'react';
import {
  View,
  Text,
  TouchableOpacity,
  Dimensions,
  StyleSheet,
  Image
} from 'react-native';
//imagenes
import Img from '../assets/img2.png'
import Colors from '../config/colors'
import Constants from '../config/constants'
import Butons from './OnBoarding/ButtonPN'
import Desc from './OnBoarding/Description'
```

Imports de
librerías y
componentes

Funcion para
la vista

```
function DefensaHito3 ({ img,text,text1,text2,onPress,onPress2 }) {
  return (
    <View style={styles.container}>
      <View style={styles.texto}>
        <Desc
          img={img}
          text={text}
          text1={text1}
          text2={text2}
        />
      </View>
      <View style={styles.boton}>
        <Butons
          onPress={onPress}
          onPress2={onPress2}
        />
      </View>
    </View>);};
```

```
const { height } = Dimensions.get("screen");  
const height_logo = height * 0.28;
```

```
const styles = StyleSheet.create({  
  container:{
```

```
    height:'100%'  
  },
```

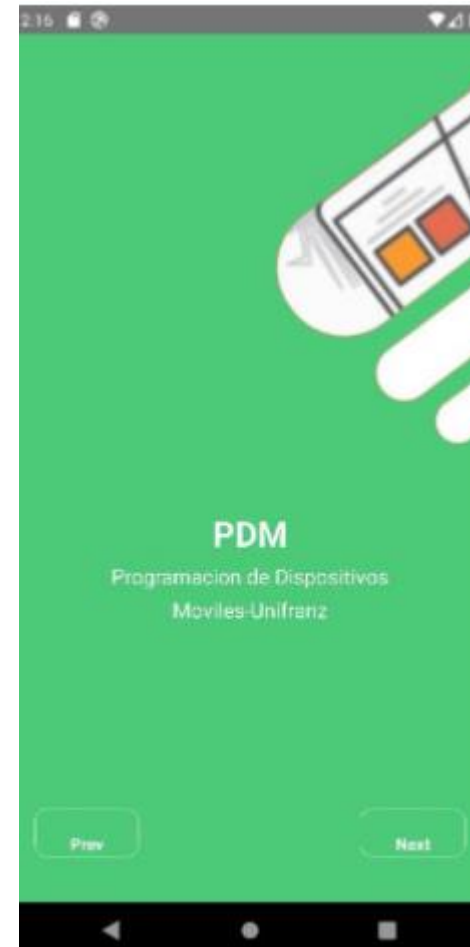
```
    texto:{
```

```
  },
```

```
    boton:{  
      paddingTop:570  
    }  
  }  
});
```

Estilos para
realizar el
diseño

**El Diseño debería
quedar algo así:**





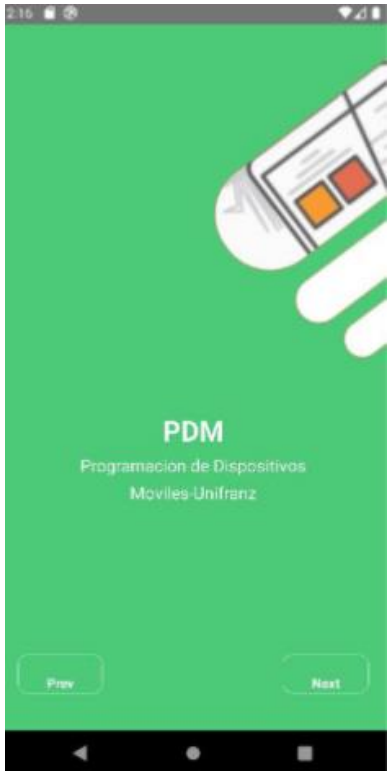
05

Vistas

Desarrollo

Requerimos 5 vistas, desde cada una de estas vistas enviaremos los props a sus respectivos componentes. Son 3 vistas para el Aplash screen usaremos 3, una vista de login y finalmente una vista para el usuario no logueado.

Vista PGM



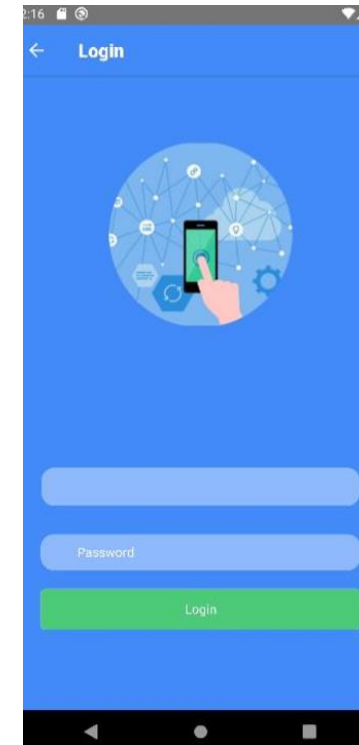
Vista Defensa



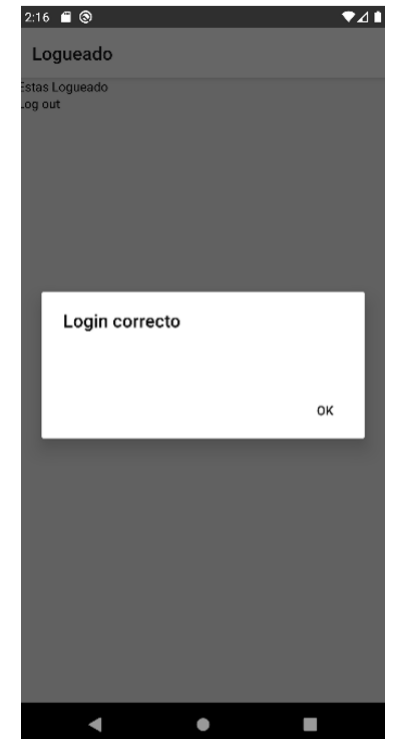
Vista Firebase



Vista Login



Vista Logueado





Componente PDM

Desarrollo

Para esta vista utilizaremos un backgroundcolor verde y enviaremos los props desde las constantes, solo será necesario importar el onBoarding.

```
import React from 'react';
import {
  View,
  Text,
  TouchableOpacity,
  Dimensions,
  StyleSheet,
  Image
} from 'react-native';
//imagenes
import Img from '../assets/img1.png'
import Colors from '../config/colors'
import colors from '../config/colors';
import SplashScreen2 from '../View/login/DefensaHito3'
import OnBoarding from '../components/OnBoardingFactory'
import Constants from '../config/constants'
```

Imports de
librerías y
componentes

Función para
la vista

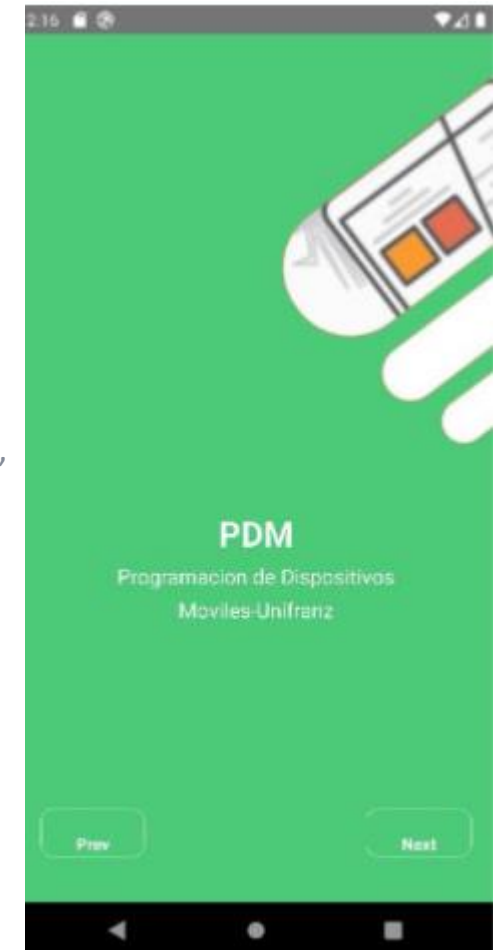
```
const PDM = ({ navigation }) => {
  return (
    <View style={styles.container}>
      <OnBoarding
        img={Img}
        text={Constants.STRINGS.IN}
        text1={Constants.STRINGS.IN2}
        text2={Constants.STRINGS.IN3}
        onPress={() => navigation.navigate('SplashScreen1')}
        onPress2={() => navigation.navigate('SplashScreen2')}
      />
    </View>
  );
};
```

```
const { height } = Dimensions.get("screen");
const height_logo = height * 0.28;
const styles = StyleSheet.create({
  container: {
    backgroundColor: Colors.green,
  }, logo: {
    width: height_logo,
    height: height_logo
  }, image: { resizeMode: 'stretch'
  }, title: {
    alignSelf: 'center',
    color: Colors.white,
    fontSize: 30,
    fontWeight: 'bold'
  }, text: {
    alignSelf: 'center',
    textAlign: 'center',
    color: Colors.white,
    fontSize: 18,
    marginTop: 5
  },
},
```

Estilos para
realizar el
diseño

```
button: {
  alignItems: 'flex-end',
  marginTop: 100,
  paddingLeft: 200
}, signIn: {
  width: '100%',
  alignItems: 'center',
  justifyContent: 'center',
  marginBottom: 12,
  paddingVertical: 12,
  borderRadius: 10,
  borderWidth: StyleSheet.hairlineWidth,
  borderColor: colors.white,
}, textSign: {
  color: Colors.white,
  textAlign: 'center',
  height: 20  }}}
```

**El Diseño debería
quedar algo así:**





05.2 Componente PDM

Desarrollo

Para esta vista utilizaremos un background color Anaranjado y enviaremos los props desde las constantes, solo será necesario importar el onBoarding.

```
import React from 'react';
import {
  View,
  Text,
  TouchableOpacity,
  Dimensions,
  StyleSheet,
  Image
} from 'react-native';
//imagenes
import Img from '../assets/img2.png'
import Colors from '../config/colors'
import Constants from '../config/constants'
import OnBoarding from '../components/OnBoardingFactory'
```

Imports de
librerías y
componentes

Función para
la vista

```
const DefensaHito3 = ({ navigation }) => {
  return (
    <View style={styles.container}>
      <OnBoarding
        img={Img}
        text={Constants.STRINGS.TD}
        text1={Constants.STRINGS.TD2}
        text2={Constants.STRINGS.TD3}
        onPress={() => navigation.navigate('SplashScreen1')}
        onPress2={() => navigation.navigate('SplashScreen3')}
      />
    </View>
  );
}
```

```
const { height } = Dimensions.get("screen");
const height_logo = height * 0.28;

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: Colors.Naranja,

  },
});
```

Estilos para
realizar el
diseño

El Diseño debería
quedar algo así:





05.3 Componente Firebase

Desarrollo

Para esta vista utilizaremos un backgroundcolor Azul y enviaremos los props desde las constantes, solo será necesario importar el onBoarding.

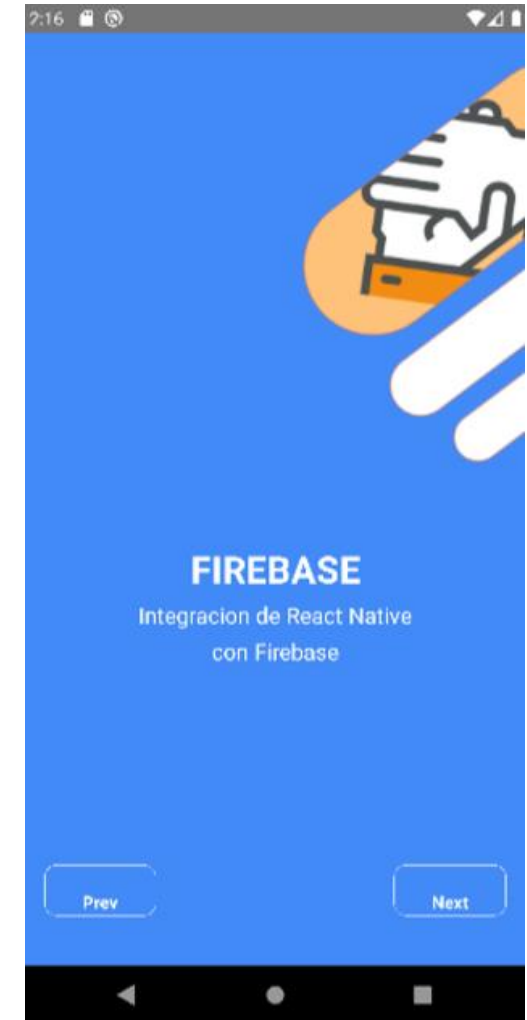
```
import React from 'react';
import {
  View,
  Text,
  TouchableOpacity,
  Dimensions,
  StyleSheet,
  Image
} from 'react-native';
//imagenes
import Img from '../assets/img1.png'
import Colors from '../config/colors'
import colors from '../config/colors';
import SplashScreen2 from '../View/login/DefensaHito3'
import OnBoarding from '../components/OnBoardingFactory'
import Constants from '../config/constants'
```

Imports de
librerías y
componentes

Función para
la vista

```
const FirebaseIntegration = ({ navigation }) => {
  return (
    <View style={styles.container}>
      <OnBoarding
        img={Img}
        text={Constants.STRINGS.FB}
        text1={Constants.STRINGS.FB2}
        text2={Constants.STRINGS.FB3}
        onPress={() => navigation.navigate('SplashScreen2')}
        onPress2={() => navigation.navigate('Login')}
      />
    </View>
  );
};
```


El Diseño debería quedar algo así:



Estilos para
realizar el
diseño

```
const { height } = Dimensions.get("screen");
const height_logo = height * 0.28;
const styles = StyleSheet.create({
  container: {
    backgroundColor: Colors.blue,
  }, logo: {
    width: height_logo,
    height: height_logo
  }, image: { resizeMode: 'stretch'
  }, title: {
    alignSelf: 'center',
    color: Colors.white,
    fontSize: 30,
    fontWeight: 'bold'
  }, text: {
    alignSelf: 'center',
    textAlign: 'center',
    color: Colors.white,
    fontSize: 18,
    marginTop: 5
  },
});
```

```
button: {
  alignItems: 'flex-end',
  marginTop: 100,
  paddingLeft: 200
}, signIn: {
  width: '100%',
  alignItems: 'center',
  justifyContent: 'center',
  marginBottom: 12,
  paddingVertical: 12,
  borderRadius: 10,
  borderWidth: StyleSheet.hairlineWidth,
  borderColor: colors.white,
}, textSign: {
  color: Colors.white,
  textAlign: 'center',
  height: 20  }}}
```



05.4 Componente Login

Desarrollo

Es bastante parecido al realizado en clases, solo cambie algunas funciones y la manera en la que implementamos react.

```
import React, { useState } from 'react';
import {
  StyleSheet, View,
  Text, SafeAreaView,
  KeyboardAvoidingView,
  TextInput, Alert
} from 'react-native';
import * as firebase from 'firebase';

import ButtonLogin from '../components/login/button';
import TextInputLogin from '../components/login/textInput';
import EmailTextField from '../components/login/emailTextField';

import LogoLogin from '../components/login/logo';
import DismissKeyboard from '../components/login/dismissKeyboard';
import Constants from '../config/constants';
import Colors from '../config/colors';
import { validateEmail } from '../utils/utills';
import colors from '../config/colors';
```

Imports de
librerías y
componentes

Función para
la vista

```
const login = ({ navigation }) => {
  const [email, setEmail] = useState("");
  const [password, setPassword] = useState("");

  const login = async () => {
    if (!email || !password) {
      Alert.alert("Todos los campos son oblogatorios");
    }
    else {
      if (!validateEmail(email)) {
        Alert.alert("El E-Mail no es valido");
      }
      else {
        await firebase
          .auth()
          .signInWithEmailAndPassword(email, password)
          .then(() => {
            Alert.alert("Login correcto");
          })
          .catch(() => {
            Alert.alert("Error al Iniciar Sesion");
          })
      }
    }
  }
}
```

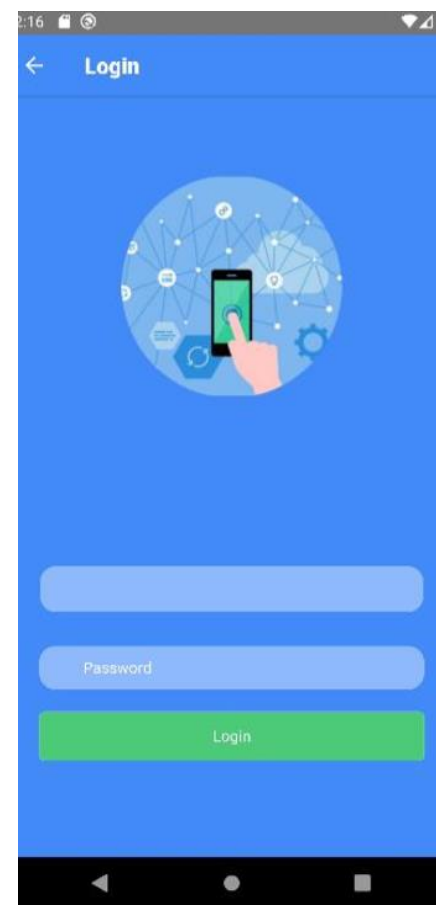
```
return (  
  <DismissKeyboard>  
    <KeyboardAvoidingView  
      style={stylesLoginScreen.container}  
      behavior="height"  
      enabled>  
    <View style={stylesLoginScreen.container}>  
      <SafeAreaView>  
        <LogoLogin style={stylesLoginScreen.logo} />  
        <View style={stylesLoginScreen.form}>  
          <EmailTextField  
            onChange={e => setEmail(e.nativeEvent.text)}  
            placeholder={Constants.STRINGS.EMAIL}  
          />  
          <TextInputLogin  
            onChange={e => setPassword(e.nativeEvent.text)}  
            placeholder={Constants.STRINGS.PASSWORD}  
            autoCapitalize="none"  
            secureTextEntry={true}  
            autoComplete={false}  
          />  
          <ButtonLogin  
            onPress={login}  
            titleButton={Constants.STRINGS.TITLE_BUTTON}  
          />  
        </View>  
      </SafeAreaView>  
    </View>  
  </KeyboardAvoidingView>  
</DismissKeyboard> );}
```

Estilos
para
realizar
el diseño

```
const stylesLoginScreen = StyleShee  
t.create({  
  container: {  
    flex: 1,  
    backgroundColor: Colors.blue,  
    alignItems: 'center',  
  
  },  
  logo: { flex: 1, width: '100%', resizeMode: 'contain', alignSelf: 'center' },  
  form: { flex: 1, justifyContent: 'center', width: '80%' },  
  texto: { fontWeight:'bold',fontSize:20 ,paddingTop:15, color:Colors.white },  
  contenedor:  
  {  
    backgroundColor: colors.blue,  
    width: '100%',  
    alignSelf:'flex-start',  
  }  
});
```

Función para
la vista

El Diseño debería
quedar algo así:





05.5 Componente Register

Desarrollo

Con el uso de la navegacion y la autentificación de firebase se redireccionara a esta vita.

```
import React, { useState } from 'react';
import {
  StyleSheet,
  View,
  Text,
  SafeAreaView,
  KeyboardAvoidingView,
  TextInput,
  Alert,
  TouchableOpacity
} from 'react-native';
import * as firebase from 'firebase';
```

Imports de
librerías y
componentes

Función para
la vista

```
const register = ({ navigation }) => {
  return (
    <View>
      <Text>Estas Logueado</Text>

      <TouchableOpacity onPress={() => firebase.auth().signOut()}>
        <Text>Log out</Text>
      </TouchableOpacity>
    </View>

  );
}
```

El Diseño debería quedar algo asi:

