

UNIVERSIDAD PRIVADA FRANZ TAMAYO



DEFENSA HITO 3 - TAREA FINAL

Nombre Completo: **Univ. Jomar Edgar Camacho Cuellar**

Asignatura: **PDM**

Carrera: **INGENIERÍA DE SISTEMAS**

Paralelo: **PDM(1)**

Docente: **Lic. William R. Barra Paredes**

fecha: **10/5/2020**

github: <https://github.com/CJJOM/pdm/tree/master/Hito3>

PARTE TEÓRICA

- Defina que es un componente en React Native y muestre un ejemplo.

Los componentes son como las funciones de JavaScript. Aceptan entradas llamadas “props” y devuelven a React elementos que describen lo que debe aparecer en la pantalla.

```
import React, { Component } from 'react';
import { StyleSheet, View, Text, Image } from 'react-native';
import Images from '../Config/Images';

export default class Logo extends Component {
  constructor(props) {
    super(props);
  }

  render(){
    return(
      <View style={styles.container}>
        <Image source={Images.LOGO} style={styles.image}></Image>
        <Text style={styles.text}>COCHABAMBA</Text>
      </View>
    );
  }
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    alignItems: 'center',
    marginTop: 50,
  },
  image: {
    width: 200,
    height: 80,
  },
  text: {
    color: 'white',
    fontWeight: 'bold',
    backgroundColor: 'transparent',
    marginTop: 20,
  },
});
```

- Explique cómo se realiza la navegación entre screens en React Native
.Se usa la propiedad Navigator que tiene la función de navigation.

```
const _onPressn = () => {
  navigation.navigate('FIREBASE');
};
```

- Que significa IaaS, PaaS y SaaS

IaaS

La infraestructura como servicio (IaaS) es una forma de computación en la nube que proporciona recursos informáticos virtualizados a través de internet.

PaaS

PaaS, abreviatura de plataforma como servicio, es un término que hace referencia a un servicio en la nube a través del cual el proveedor proporciona al cliente un entorno de desarrollo

SaaS

Un sistema SaaS o Software as a Service, es un modelo de distribución de software en el que tanto el software como los datos manejados son centralizados y alojados en un único servidor externo a la empresa.

- Que es Firebase, Firestore y explique a que se refiere cuando se habla de Baas.

Firebase

Firebase es la nueva y mejorada plataforma de desarrollo móvil en la nube de Google. Se trata de una plataforma disponible para diferentes plataformas (Android, iOS, web), con lo que de esta forma presentan una alternativa seria a otras opciones para ahorro de tiempo en el desarrollo como Xamarin.

Firestore

Es la base de datos más reciente de Firebase para el desarrollo de apps para dispositivos móviles. Aprovecha lo mejor de Realtime Database con un modelo de datos nuevo y más intuitivo.

BaaS

El servicio de respaldo como servicio (BaaS) de CATBackup le ofrece la capacidad de realizar copias de seguridad de datos en una infraestructura compartida basada en la nube. Se adapta a sus necesidades de datos y nunca tendrá que preocuparse por la cantidad de cintas necesarias.

- Defina o explique si React es lo mismo que React Native. Si son distintos liste cuales son las diferencias.

REACT

ReactJS es una librería JavaScript, desarrollada y mantenida por Facebook, enfocada a la visualización. Como React se encarga sólo de la vista de la aplicación dentro de un paradigma Modelo-Vista-Controlador, desde Facebook recomiendan encarecidamente el uso de ReactJS con Flux, que es grosso modo, un patrón de diseño software. Por este motivo, muchas veces vemos en combinación a React JS y Redux.

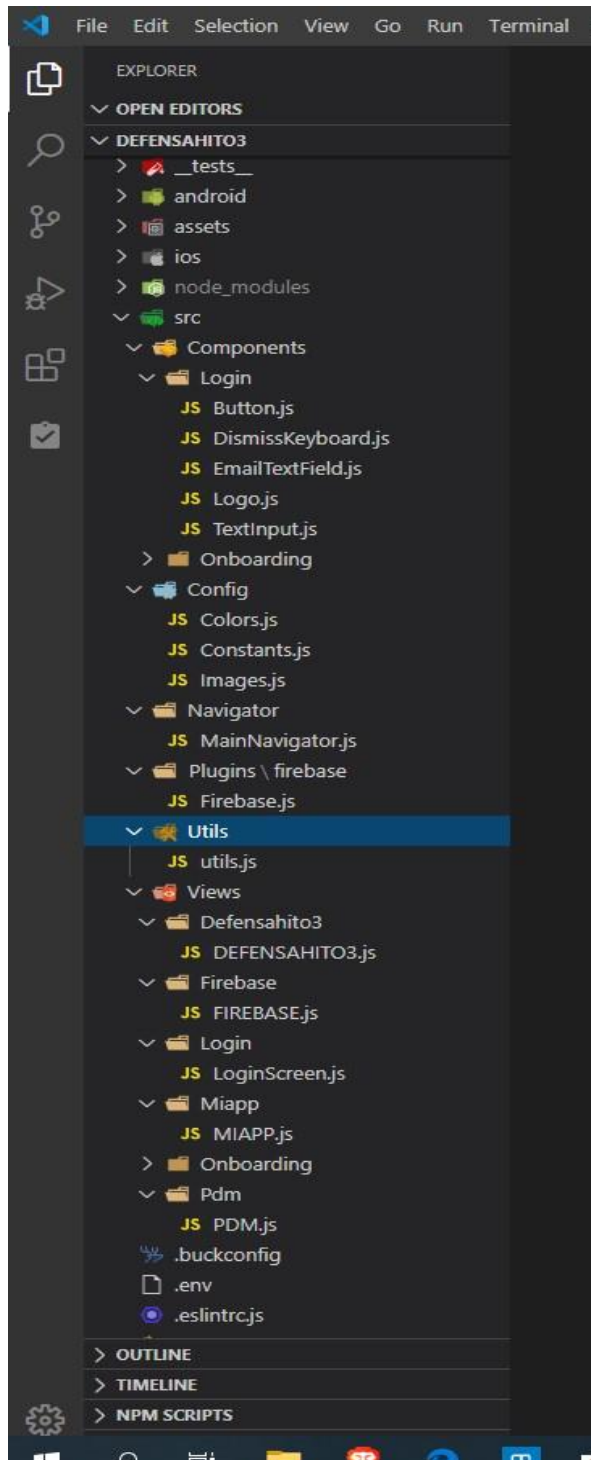
REACT NATIVE

Las vistas las crea de forma nativa. Esto significa que tiene un intérprete que a medida que la app se va ejecutando. Lo lee el *HTML* de estas vistas y va colocando cada uno de los componentes nativos en su posición.

PARTE PRÁCTICA

Directorio

Creación de un nuevo proyecto en FIREBASE.



Configuración de React native con Firebase para la integración.

```
import firebase from 'firebase/app';
import 'firebase/firestore';
import 'firebase/auth';
import 'firebase/storage';
/*import {
  API_KEY,
  AUTH_DOMAIN,
  DATABASE_URL,
  PROJECT_ID,
  STORAGE_BUCKET,
  MESSAGING_SENDER_ID,
  APP_ID
} from 'react-native-dotenv';*/

const firebaseConfig = {
  apiKey: "AIzaSyC5shTWh8hHs6YtfJmCLlED_rTMHMx5Nyw",
  authDomain: "defensah3pdm.firebaseio.com",
  databaseURL: "https://defensah3pdm.firebaseio.com",
  projectId: "defensah3pdm",
  storageBucket: "defensah3pdm.appspot.com",
  messagingSenderId: "961127200611",
  appId: "1:961127200611:web:eeaec4d1b609f69245c556"
};

const fire = firebase.initializeApp(firebaseConfig);
//export const firestore = firebase.firestore();
export default fire;
```

Creación de el componente PDM (1ra vista)

```
import React, { useState } from 'react';
import { StyleSheet, View, SafeAreaView, KeyboardAvoidingView, Alert, Text, Image } from 'react-native';

import Button_pn from '../../Components/Login/Button';

import Images from '../../Config/Images';
import Constants from '../../Config/Constants';
import Colors from '../../Config/Colors';

const PdmScreen = ({navigation}) => {
```

```

const _onPressn = () => {
  navigation.navigate('DEFENSAHITO3');
};

return (
  <View style={styles.container}>
    <Image
      style={styles.imagen}
      source={Images.FPDM}
    ></Image>
    <Text style={styles.text1}>{Constants.STRING.TITLE_PDM}</Text
  >
    <Text style={styles.text}>Programación de Dispositivos Mobile
s - Unifranz</Text>
    <View style={styles.containerbutton}>
      <Button_pn
        titleButton={Constants.STRING.BUTTON_P}
      >
      </Button_pn>
      <Button_pn
        onPress={_onPressn}
        titleButton={Constants.STRING.BUTTON_N}
      >
      </Button_pn>
    </View>
  </View>
);
};

const styles = StyleSheet.create({
  container: {
    flex: 1,
    flexDirection: 'column',
    backgroundColor: Colors.green,
  },
  text:{
    color: Colors.white,
    textAlign: 'center',
    justifyContent: 'center',
    alignItems: 'center',
    fontSize: 30,
    paddingBottom: 270
  },
  text1:{
    textAlign:'center',
    paddingBottom: 20,
    color: Colors.white,
    fontSize: 30,
    fontWeight: 'bold',

```

```
        paddingBottom: 20
      },
      containerbutton: {
        flex: 0,
        justifyContent: 'space-between',
        flexDirection: 'row',
        paddingRight: 10,
        paddingLeft: 10
      },
      imagen: {
        width: '100%',
        height: 450,
        marginTop: -70,
        marginLeft: 100
      }
    }
  });

export default PdmScreen;
```




Creación del componente DEFENSAHITO3 (2da Vista)

```
import React, { useState } from 'react';
import { StyleSheet, View, SafeAreaView, KeyboardAvoidingView, Alert, Text, Image } from 'react-native';

import Button_pn from '../../Components/Login/Button';
import Images from '../../Config/Images';
import Constants from '../../Config/Constants';
import Colors from '../../Config/Colors';

const Defensah3Screen = ({ navigation }) => {
  const _onPressn = () => {
    navigation.navigate('FIREBASE');
  };

  const _onPressp = () => {
    navigation.navigate('PDMScreen');
  };

  return (
    <View style={styles.container}>
      <Image
        style={styles.imagen}
        source={Images.FDEFH3}
      >
      </Image>
      <Text style={styles.text1}>{Constants.STRING.DEF_H3}</Text>
      <Text style={styles.text}>Univ: Jomar Camacho{"\n"}Gestión 2020</Text>
    </View>
    <View style={styles.containerbutton}>
      <Button_pn
        onPress={_onPressp}
        titleButton={Constants.STRING.BUTTON_P}
      >
      </Button_pn>
      <Button_pn
        onPress={_onPressn}
        titleButton={Constants.STRING.BUTTON_N}
      >
      </Button_pn>
    </View>
  </View>
  );
};

const styles = StyleSheet.create({
  container: {
    flex: 1,
```

```
        backgroundColor: Colors.orange
      },
      text: {
        width: 380,
        textAlign: 'center',
        color: Colors.white,
        justifyContent: 'center',
        alignItems: 'center',
        fontSize: 30,
        paddingLeft: 20,
        paddingBottom: 230
      },
      text1: {
        textAlign: 'center',
        color: Colors.white,
        fontSize: 40,
        fontWeight: 'bold',
        paddingTop: 30,
        paddingBottom: 20
      },
      containerbutton: {
        flex: 0,
        justifyContent: 'space-between',
        flexDirection: 'row',
        paddingRight: 10,
        paddingLeft: 10,
      },
      imagen: {
        width: '100%',
        height: 450,
        marginTop: -70,
        marginLeft: 100
      }
    }
  ));

export default Defensah3Screen;
```

8:19

LTE



DEFENSA HITO 3

Univ: Jomar Camacho Gestion
2020

Prev

Next

— ×



Creación del componente FIREBASE (3ra Vista)

```
import React, { useState } from 'react';
import { StyleSheet, View, SafeAreaView, KeyboardAvoidingView, Alert, Text, Image } from 'react-native';

import Button_pn from '../../Components/Login/Button';

import Images from '../../Config/Images';
import Constants from '../../Config/Constants';
import Colors from '../../Config/Colors';

const FirebaseScreen = ({navigation}) => {
  const _onPressn = () => {
    navigation.navigate('Login');
  };

  const _onPressp = () => {
    navigation.navigate('DEFENSAHITO3');
  };

  return (
    <View style={styles.container}>
      <Image
        style={styles.imagen}
        source={Images.FFIRE}
      ></Image>
      <Text style={styles.text1}>{Constants.STRING.FIRE}</Text>
      <Text style={styles.text}>
        Integración de React Native con Firebase
      </Text>
      <View style={styles.containerbutton}>
        <Button_pn
          onPress={_onPressp}
          style={styles.boton1}
          titleButton={Constants.STRING.BUTTON_P}
        >
      </Button_pn>
        <Button_pn
          onPress={_onPressn}
          titleButton={Constants.STRING.BUTTON_N}
        >
      </Button_pn>
      </View>
    </View>
  );
};
```

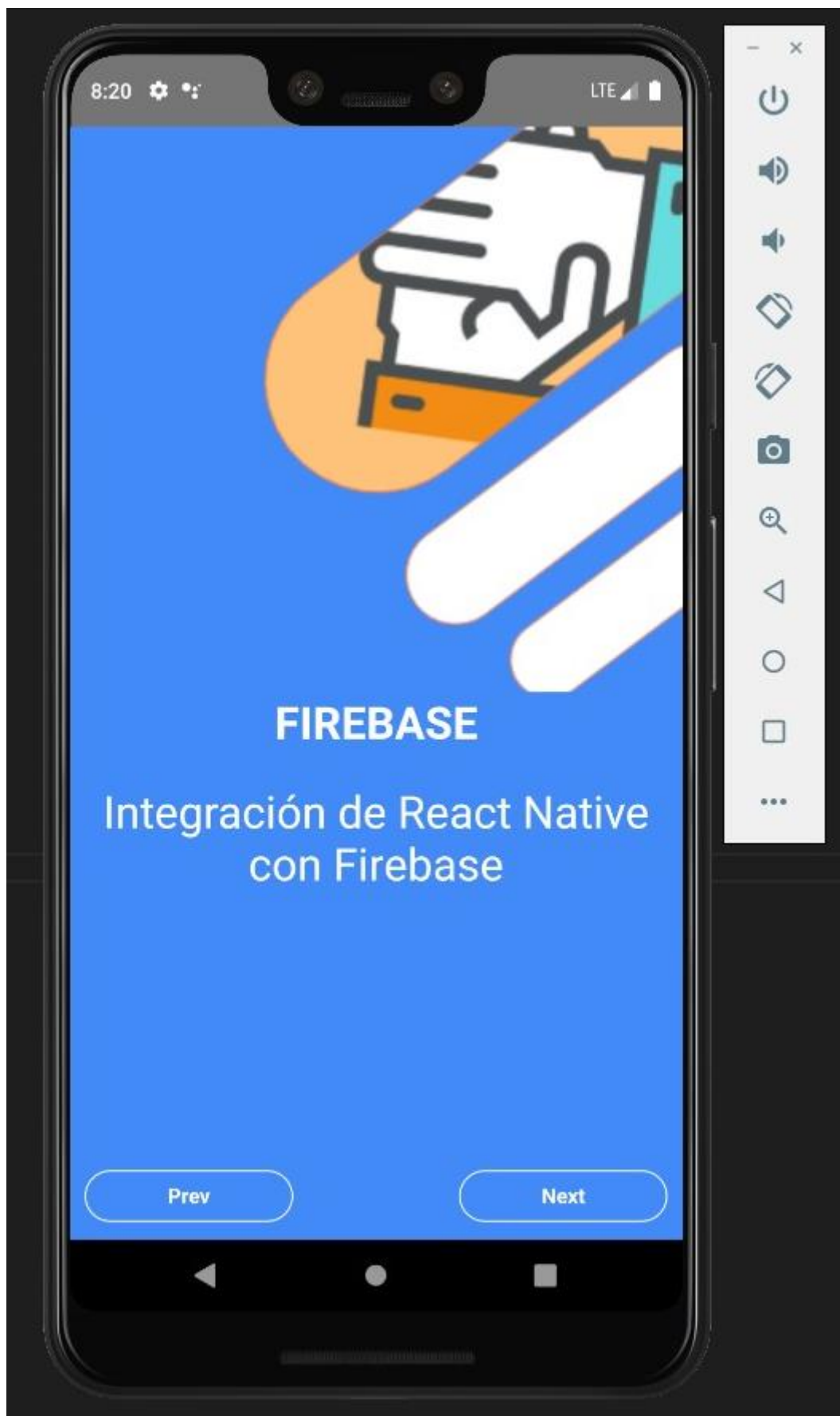
```

});

const styles = StyleSheet.create({
  container: {
    flex: 1,
    flexDirection: 'column',
    backgroundColor: Colors.blue,
  },
  text: {
    width: 380,
    textAlign: 'center',
    color: Colors.white,
    justifyContent: 'center',
    alignItems: 'center',
    fontSize: 30,
    paddingLeft: 20,
    paddingBottom: 230
  },
  text1: {
    textAlign: 'center',
    color: Colors.white,
    fontSize: 40,
    fontWeight: 'bold',
    paddingTop: 30,
    paddingBottom: 20
  },
  containerbutton: {
    flex: 0,
    justifyContent: 'space-between',
    flexDirection: 'row',
    paddingRight: 10,
    paddingLeft: 10,
  },
  imagen: {
    width: '100%',
    height: 450,
    marginTop: -70,
    marginLeft: 100
  }
});

export default FirebaseScreen;

```



Creación del componente LOGIN (4ta vista)

```
import React, { useState } from 'react';
import { StyleSheet, View, SafeAreaView, KeyboardAvoidingView, Alert } from 'react-native';

import ButtonLogin from '../../Components/Login/Button';
import TextInputLogin from '../../Components/Login/TextInput';
import LogoLogin from '../../Components/Login/Logo';
import EmailTextField from '../../Components/Login/EmailTextField';
import DismissKeyboard from '../../Components/Login/DismissKeyboard';
import FirebasePlugin from '../../Plugins/firebase/Firebase';

import Utils from '../../Utils/Utils';
import Images from '../../Config/Images';
import Constants from '../../Config/Constants';
import Colors from '../../Config/Colors';

const LoginScreen = ({navigation}) => {
  const [email, setEmail] = useState('');
  const [errorEmail, setErrorEmail] = useState('');
  const [password, setPassword] = useState('');
  const [errorPassword, setErrorPassword] = useState('');

  const _validateEmailAddress = () => {
    let isValidEmail = Utils.isValidEmail(email);
    isValidEmail ? setErrorEmail('') : setErrorEmail(Constants.STRING.EMAIL_ERROR);
    return isValidEmail;
  }

  const _validatePassword = () => {
    let isValidPassword = Utils.isValidField(password);
    isValidPassword ? setErrorPassword('') : setErrorPassword(Constants.STRING.USERNAME_ERROR);
    return isValidPassword;
  }

  const _onPress = () => {
    //console.log('press button!!!');
    let emailData = _validateEmailAddress();
    let passwordData = _validatePassword();

    if (emailData && passwordData) {
      LoginApp(email, password);
    } else {
      Alert.alert('valores no aceptados!!');
    }
  }
}
```



```

const LoginApp = (email, password) => {
  try {
    FirebasePlugin.auth().signInWithEmailAndPassword(email, password)
      .then((user) => {
        Alert.alert('Usuario logeado');
      })
      .catch((error) => {
        FirebasePlugin.auth().createUserWithEmailAndPassword(email, pas
sword)
          .then((user) => {
            debugger;
            navigation.navigate('Miapp');
            Alert.alert('Usuario Creado');
          })
          .catch((error) => {
            Alert.alert(error.message);
          })
        })
      } catch (error) {
        Alert.alert(error.message);
      }
  }

  return (
    <DismissKeyboard>
    <KeyboardAvoidingView style={stylesLoginScreen.container} behavior=
"height" enabled>
    <View style={stylesLoginScreen.container}>
      <SafeAreaView>
        <LogoLogin style={stylesLoginScreen.logo} />
        <View style={stylesLoginScreen.form}>
          <EmailTextField
            onChangeText={(email) => { setEmail(email) }}
            onEndEditing={_validateEmailAddress}
            source={Images.EMAIL}
            error={errorEmail}
            placeholder={Constants.STRING.EMAIL}
            secureTextEntry={false}
            autoCorrect={false}>
        </EmailTextField>
        <TextInputLogin
            onChangeText={(password) => { setPassword(password) }}
            onEndEditing={_validatePassword}
            source={Images.PASSWORD}
            error={errorPassword}
            placeholder={Constants.STRING.PASSWORD}
            secureTextEntry={true}
            autoCorrect={false}>

```

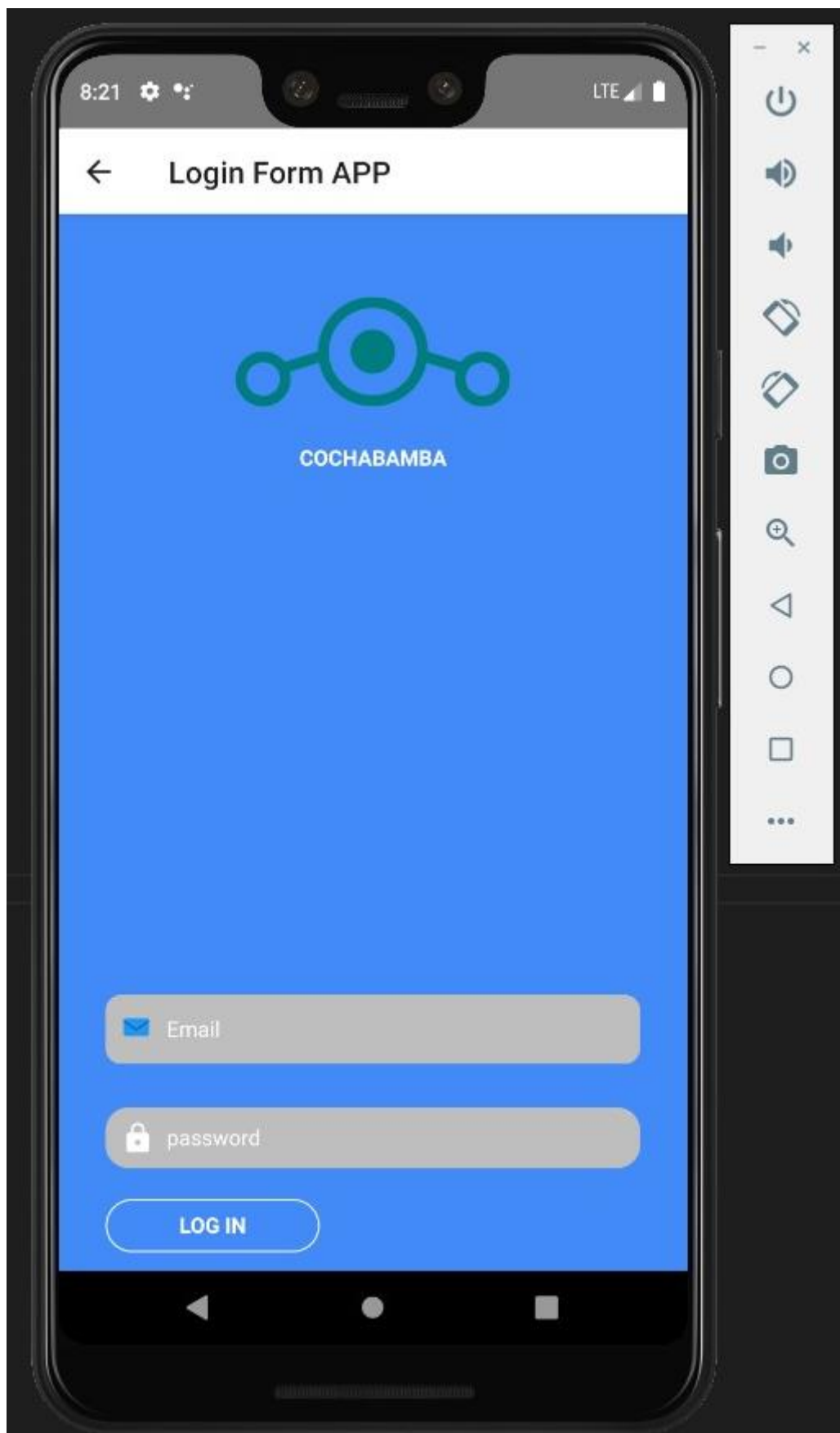
```

        </TextInputLogin>
        <ButtonLogin
            onPress={_onPress}
            titleButton={Constants.STRING.TITLE_BUTTON}>
        </ButtonLogin>
    </View>
</SafeAreaView>
</View>
</KeyboardAvoidingView>
</DismissKeyboard>
);
}

const stylesLoginScreen = StyleSheet.create({
    container: {
        flex: 1,
        backgroundColor: Colors.blue,
        alignItems: 'center',
    },
    logo: { width: '100%', resizeMode: 'contain', alignSelf: 'center' },
    form: { justifyContent: 'center', width: '80%' },
});

export default LoginScreen;

```



Creacion del componente MIAPP (Vista que se mostrara despues de hacer LOGIN).

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

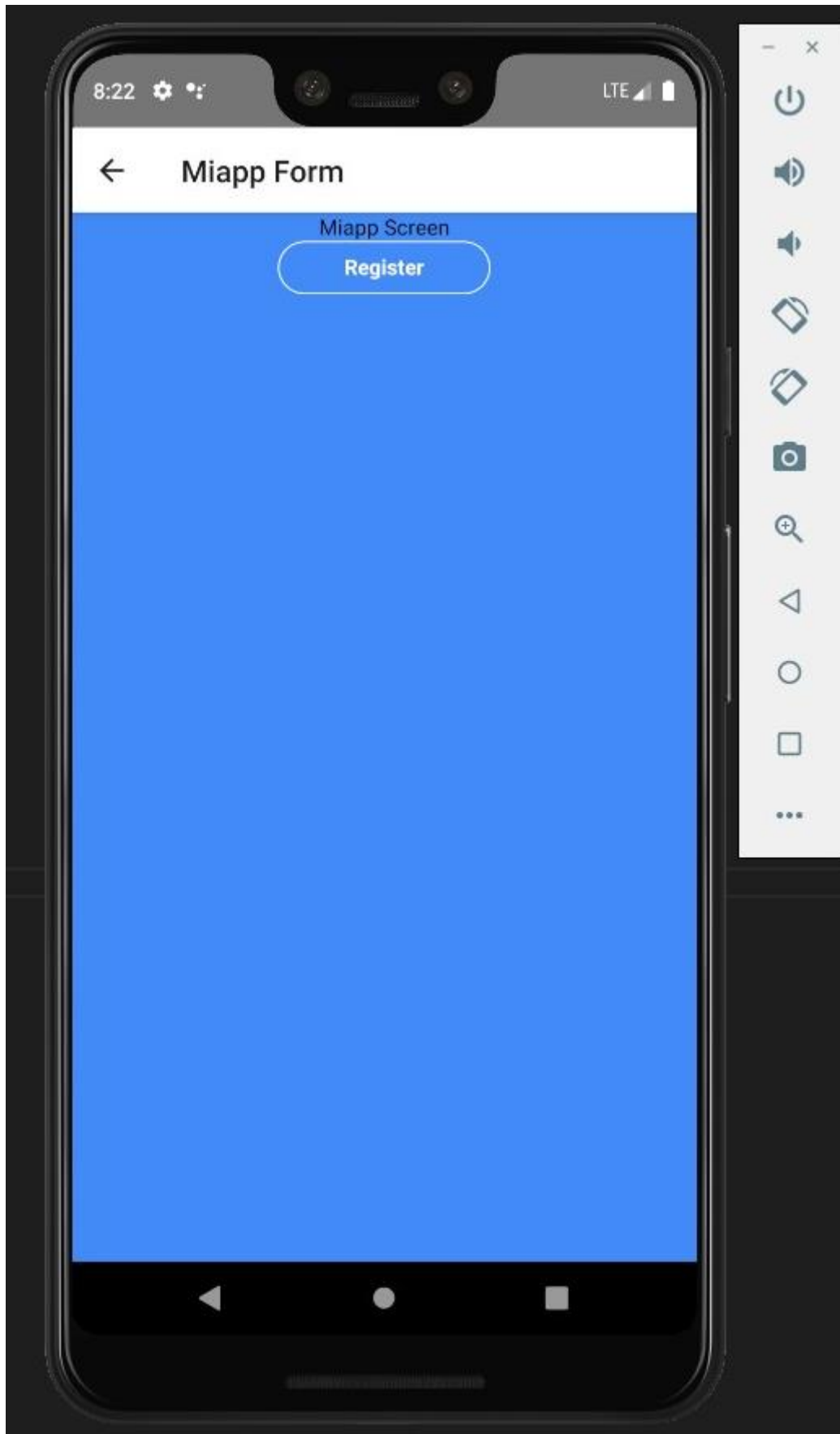
import Constants from '../../Config/Constants';
import Colors from '../../Config/Colors';
import ButtonLogin from '../../Components/Login/Button';

const MiappScreen = ({navigation}) => {
  const onPress = () => {
    console.log('register');
  };

  return (
    <View style={styles.container}>
      <Text>Miapp Screen</Text>
      <ButtonLogin onPress={onPress} titleButton={Constants.STRING.REGIST
ER} />
    </View>
  );
};

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: Colors.blue,
    alignItems: 'center',
  },
  text: {
    color: Colors.white,
    textAlign: 'center',
    fontWeight: 'bold',
    height: 20,
  },
});

export default MiappScreen;
```



COMPONENTES NECESARIOS

BUTTON

```
import React, { Component } from 'react';
import { StyleSheet, View, Text, TouchableOpacity } from 'react-native';
import Colors from '../../Config/Colors';

export default class button extends Component {

  constructor(prop) {
    super(prop);
  }

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

const stylesButton = StyleSheet.create({
  container: {
    width: 140,
    justifyContent: 'center',
    alignItems: 'center',
    alignContent: 'center',
    backgroundColor: 'transparent',
    marginBottom: 12,
    paddingVertical: 12,
    borderRadius: 50,
  },
});
```

```

        borderWidth: StyleSheet.hairlineWidth,
        borderColor: Colors.white,
        height: 35,
        borderWidth: 1,
      },
      text: {
        color: Colors.white,
        textAlign: 'center',
        height: 20,
        fontWeight: 'bold'
      },
    },
  ));

```

DISMISSKEYBOARD

```

import React from 'react';
import { TouchableWithoutFeedback, Keyboard } from 'react-native';

const DismissKeyboard = ({ children }) => {
  return (
    <TouchableWithoutFeedback onPress={() => Keyboard.dismiss()}>
      {children}
    </TouchableWithoutFeedback>
  );
}

export default DismissKeyboard;

```

EMAILTEXTFIELD

```

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

import Colors from '../../Config/Colors';
import Constants from '../../Config/Constants';

const EmailTextField = ({ onChangeText, placeholder, error, onEndEditing,
  secureTextEntry, autoCorrect, source }) => {
  return (
    <View>
      <Text style={styles.errorText}>{error}</Text>
      <View style={styles.textFieldView}>
        <Image style={styles.inlineImg} source={source}/>
        <TextInput
          style={styles.textField}
          onChangeText={onChangeText}

```

```

        onEndEditing={onEndEditing}
        placeholder={placeholder}
        secureTextEntry={secureTextEntry}
        autoCorrect={autoCorrect}
        selectionColor={Colors.blue}
        placeholderTextColor={Colors.white}
        underlineColorAndroid="transparent"
    >

    </TextInput>
  </View>
</View>
);
}

const styles = StyleSheet.create({
  textField: {
    fontSize: 14,
    flex: 1,
    marginHorizontal: 20,
    paddingLeft: 20
  },
  textFieldView: {
    height: Constants.CONFIG.HEADER_HEIGHT * 0.06,
    width: Constants.CONFIG.SCREEN_WIDTH * 0.85,
    borderRadius: 10,
    marginTop: 5,
    marginBottom: 10,
    justifyContent: 'center',
    backgroundColor: Colors.silver,
  },
  inlineImg: {
    position: 'absolute',
    zIndex: 99,
    width: 20,
    height: 20,
    left: 10,
    top: 11,
  },
  errorText: {
    fontSize: 12,
    color: Colors.red,
    marginBottom: -5,
    marginHorizontal: 20,
  },
});

export default EmailTextField;

```


LOGO

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

import Images from '../../Config/Images';

export default class Logo extends Component {
  constructor(props) {
    super(props);
  }

  render(){
    return(
      <View style={styles.container}>
        <Image source={Images.LOGO} style={styles.image}></Image>
        <Text style={styles.text}>COCHABAMBA</Text>
      </View>
    );
  }
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    alignItems: 'center',
    marginTop: 50,
  },
  image: {
    width: 200,
    height: 80,
  },
  text: {
    color: 'white',
    fontWeight: 'bold',
    backgroundColor: 'transparent',
    marginTop: 20,
  },
});
```

TEXTINPUT

```
import React, { Component } from 'react';
import { StyleSheet, View, Text, TextInput, Image } from 'react-native';
import PropTypes from 'prop-types';
```

```

import Colors from '../..//Config/Colors';

export default class TextInputForm extends Component {

  constructor(props) {
    super(props)
  }

  render() {
    return (
      <View>
        <Text style={stylesTextInput.errorText}>{this.props.error}
      </Text>
      <View>
        <Image
          style={stylesTextInput.inlineImg}
          source={this.props.source}>
        </Image>
        <TextInput
          onChangeText={this.props.onChangeText}
          onEndEditing={this.props.onEndEditing}
          style={stylesTextInput.textInput}
          selectionColor={Colors.blue}
          placeholder={this.props.placeholder}
          secureTextEntry={this.props.secureTextEntry}
          autoComplete={this.props.autoCorrect}
          placeholderTextColor={Colors.white}
          underlineColorAndroid="transparent"
        ></TextInput>
      </View>
    </View>
  );
}

TextInputForm.propTypes = {
  source: PropTypes.number.isRequired,
  placeholder: PropTypes.string.isRequired,
  autoComplete: PropTypes.bool,
  secureTextEntry: PropTypes.bool,
};

const stylesTextInput = StyleSheet.create({
  textInput: {
    backgroundColor: Colors.silver,
    alignItems: 'center',
  }
});

```

```
        height: 40,
        borderColor: '#BEBEBE',
        paddingLeft: 40,
        borderRadius: 15,
        borderBottomWidth: StyleSheet.hairlineWidth,
        marginBottom: 20,
      },
      inlineImg: {
        position: 'absolute',
        zIndex: 99,
        width: 22,
        height: 22,
        left: 10,
        top: 9,
      },
      errorText: {
        fontSize: 14,
        color: Colors.red,
        marginHorizontal: 20,
      },
    },
  ));
```