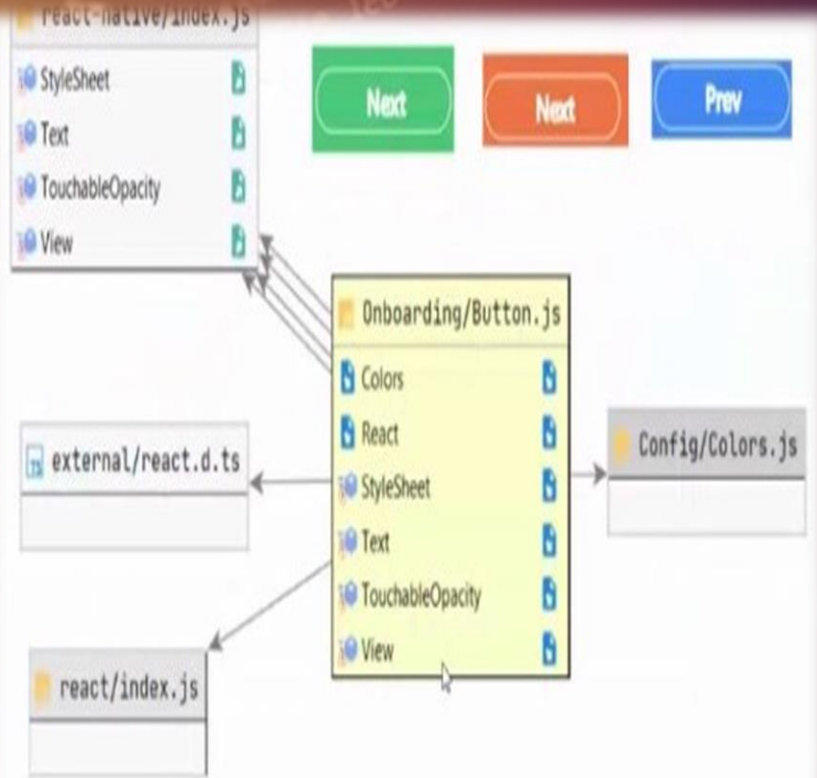




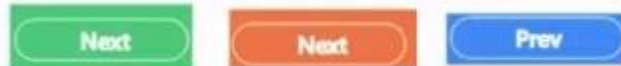
RESOLUCIÓN

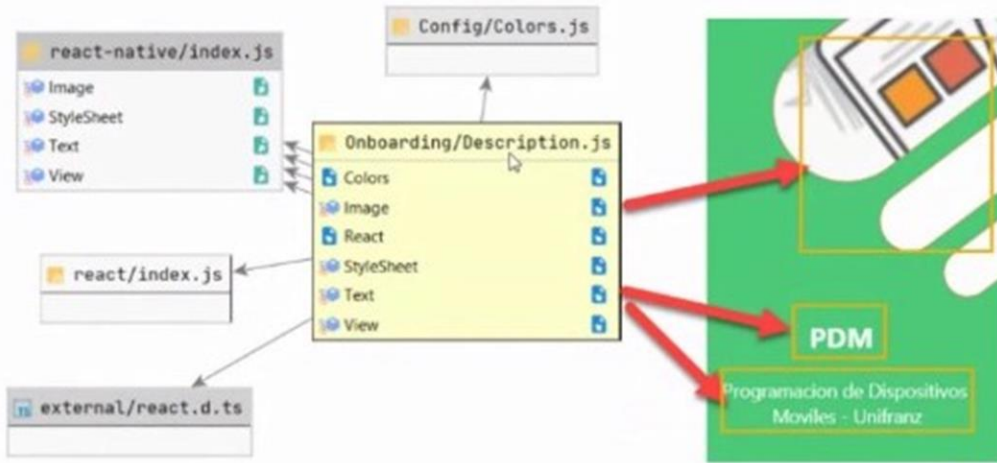
DEFENSA – HITO 3

CREAR EL COMPONENTE “BUTTON” PARA LAS ACCIONES PREV Y NEXT



```
src > Components > Onboarding > JS Button.js > stylesButton > container
1  import React, { Component } from 'react';
2  import { StyleSheet, View, Text, TouchableOpacity } from 'react-native';
3  import Colors from '../Config/Colors';
4
5
6  export default class button extends Component {
7
8      constructor(prop) {
9          super(prop);
10     }
11
12
13
14     render() {
15         return (
16             <View style={stylesButton.container}>
17                 <TouchableOpacity
18                     style={stylesButton.button}
19                     onPress={this.props.onPress}>
20
21                     <Text
22                         style={stylesButton.text}>{this.props.titleButton}</Text>
23
24                 </TouchableOpacity>
25             </View>
26         );
27     }
28 }
29
30
```





CREAR EL COMPONENTE “DESCRIPCIÓN.JS”

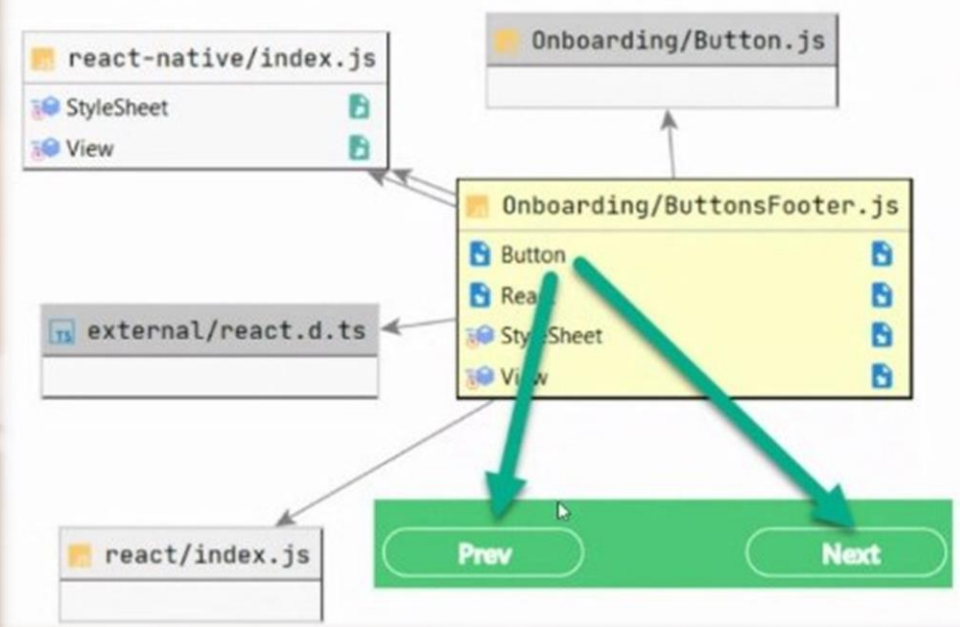
```

JS MainNavigator.js JS OnboardingFactory.js JS Description.js X JS FIREBASE.js JS
src > Components > Onboarding > JS Description.js > Description > render
1  import React, { Component } from 'react';
2  import { StyleSheet, View, Text, Image } from 'react-native';
3
4  import Colors from '../Config/Colors';
5
6
7  export default class Description extends Component {
8    constructor(prop) {
9      super(prop);
10   }
11
12   render() {
13     return (
14       <View style={styles.container}>
15         <Image
16           style={styles.imagen}
17           source={this.props.imagenes}
18         ></Image>
19         <Text style={styles.text1}>{this.props.title1}</Text>
20         <Text style={styles.text}>{this.props.desc1}</Text>
21       </View>
22     );
23   }
24 }
25
  
```

Inserción de
imagen y texto
del
componente

PDM
Programación de Dispositivos
Moviles - Unifranz

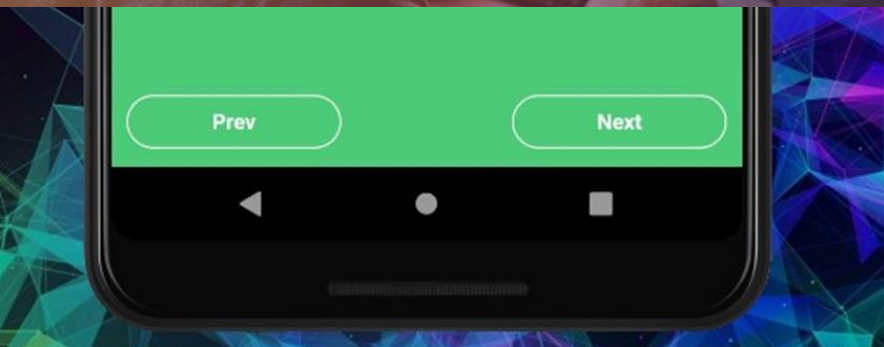
Crear el componente **ButtonsFooter.js**



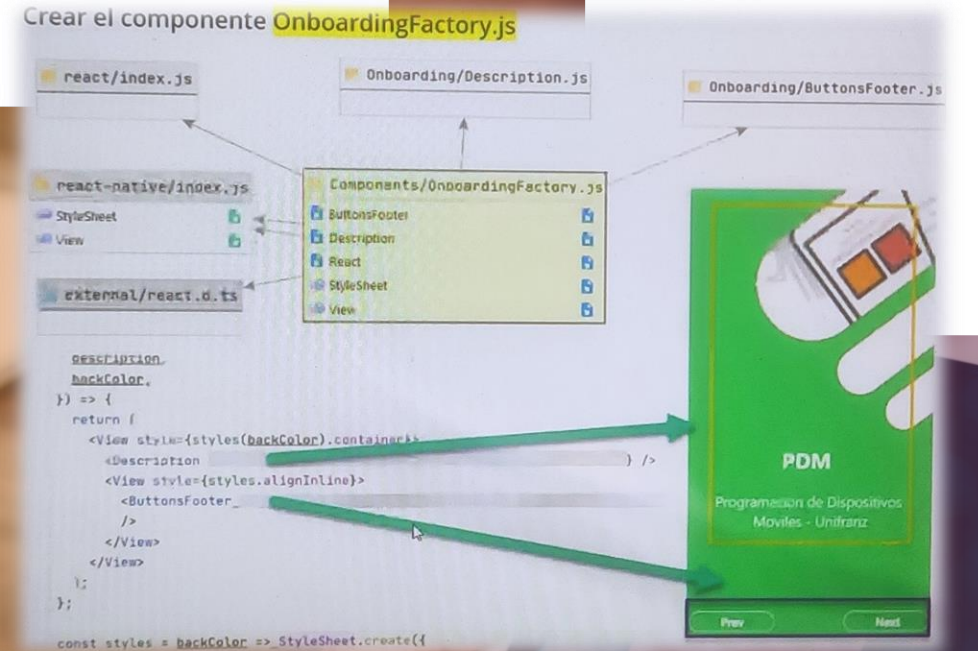
CREAR EL COMPONENTE “BUTTONSFOOTER.JS”

```
JS MainNavigator.js JS OnboardingFactory.js JS Description.js JS ButtonsFooter.js X
src > Components > Onboarding > JS ButtonsFooter.js > [0] stylesButtonsFooter > containerbutton
1  import React, { Component } from 'react';
2  import { StyleSheet, View } from 'react-native';
3
4  import Boton from '../Onboarding/Button';
5  import Constants from '../../Config/Constants';
6
7  const ButtonsFooter = ({onPressp, onPressn}) => {
8
9
10
11
12
13    return(
14      <View style={stylesButtonsFooter.containerbutton}>
15        <Boton
16          onPress={onPressp}
17          titleButton={Constants.STRING.BUTTON_P}
18        >
19        </Boton>
20        <Boton
21          onPress={onPressn}
22          titleButton={Constants.STRING.BUTTON_N}
23        >
24        </Boton>
25      </View>
26    );
27  };
28
29  const stylesButtonsFooter = StyleSheet.create({
30    containerbutton: {
31      position: 'absolute',
32      width: '100%',
33      flex: 0,
34      justifyContent: 'space-between',
35      flexDirection: 'row',
36      paddingRight: 10,
37      paddingLeft: 10,
38    }
39  });
40
41  export default ButtonsFooter;
```

Separación de 2 botones en horizontal



CREAR EL COMPONENTE "ONBOARDING.JS"

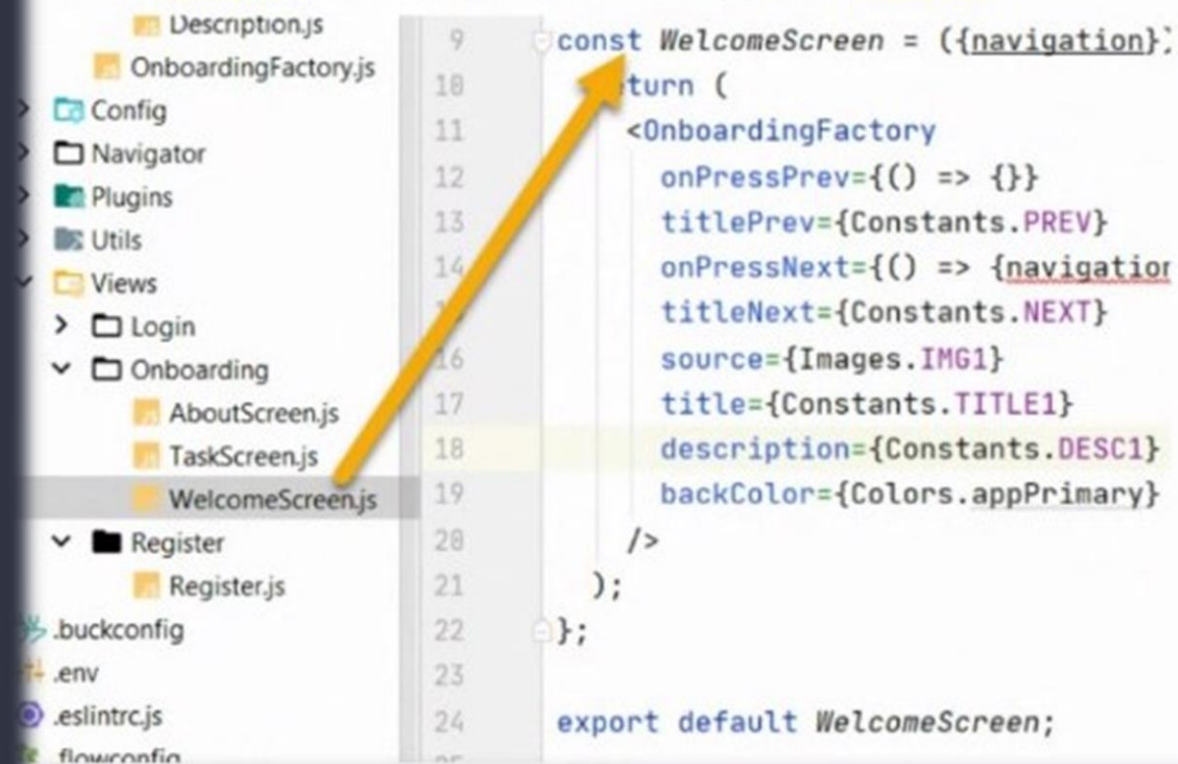


Ordenamiento en vertical de ambos componentes

```
JS MainNavigator.js JS OnboardingFactory.js X JS Description.js JS ButtonsFooter.js JS FIREBASE.js
src > Components > Onboarding > JS OnboardingFactory.js > OnboardingFactory
1  import React, { Component } from 'react';
2  import { StyleSheet, View } from 'react-native';
3
4  import ButtonsFooter from '../Onboarding/ButtonsFooter';
5  import Description from '../Onboarding/Description';
6
7
8  const OnboardingFactory = ({imagenes, title1, des1,onPressp,onPressn}) => {
9
10
11    return (
12      <View style={styleOnboardFacto.container}>
13        <Description
14          imagenes={imagenes}
15          title1={title1}
16          desc1={des1}
17        ></Description>
18        <ButtonsFooter
19          onPressp={onPressp}
20          onPressn={onPressn}
21        ></ButtonsFooter>
22      </View>
23    );
24
25  };
26
27  const styleOnboardFacto = StyleSheet.create({
28    container: {
29      flex: 1,
30      flexDirection: 'column',
31      justifyContent: 'flex-end'
32    }
33  });
34  export default OnboardingFactory;
```


CREAR LOS “SCREENS” PARA EL NAVIGATOR Y VERIFICAR EL LOGIN A FIREBASE

Crear los **SCREENS** para el navigator y verificar el login a **FIREBASE**.



```
9  const WelcomeScreen = ({navigation}):
10
11  return (
12    <OnboardingFactory
13      onPressPrev={() => {}}
14      titlePrev={Constants.PREV}
15      onPressNext={() => {navigation}}
16      titleNext={Constants.NEXT}
17      source={Images.IMG1}
18      title={Constants.TITLE1}
19      description={Constants.DESC1}
20      backgroundColor={Colors.appPrimary}
21    />
22  );
23
24  export default WelcomeScreen;
```

WelcomeScreen

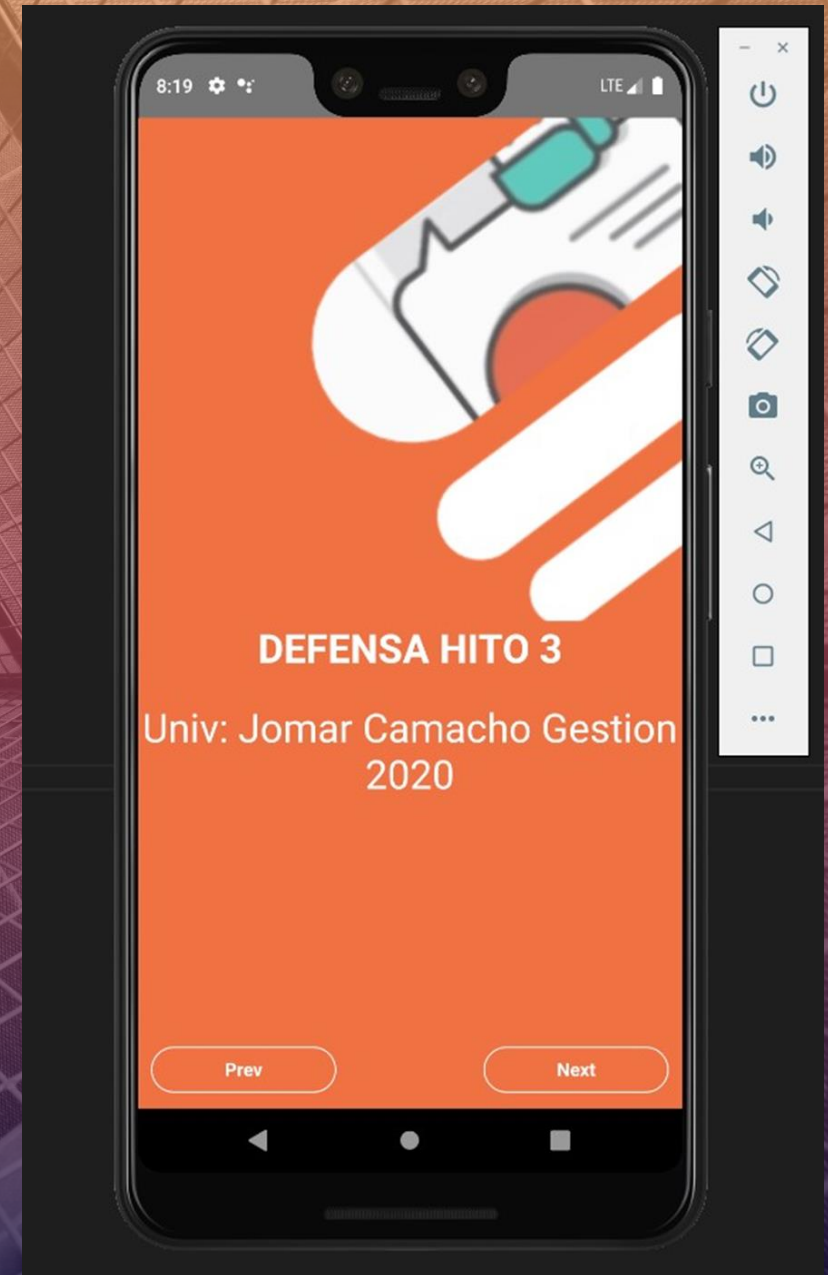
```
MainNavigator.js  JS OnboardingFactory.js  JS WelcomeScreen.js X  JS Description.js  JS Buti
> Views > Onboarding > JS WelcomeScreen.js > [🔍] default
1  import React from 'react';
2  import { StyleSheet, View } from 'react-native';
3  import Constants from '../../Config/Constants';
4  import Images from '../../Config/Images';
5
6  import OnboardingFactory from '../../Components/Onboarding/OnboardingFactory';
7
8
9  const WelcomeScreen = ({navigation}) => {
10    return(
11      <OnboardingFactory
12        imagenes={Images.FPDM}
13        title1={Constants.STRING.TITLE_PDM}
14        des1={Constants.STRING.DESC1}
15        onPressn={()=>{navigation.navigate('TaskScreen')}}
16        onPressp={()=>{navigation.navigate('WelcomeScreen')}}
17      >
18      </OnboardingFactory>
19    );
20  };
21
22  export default WelcomeScreen;
```



Llamada de todos los componentes en el screen

TaskScreen

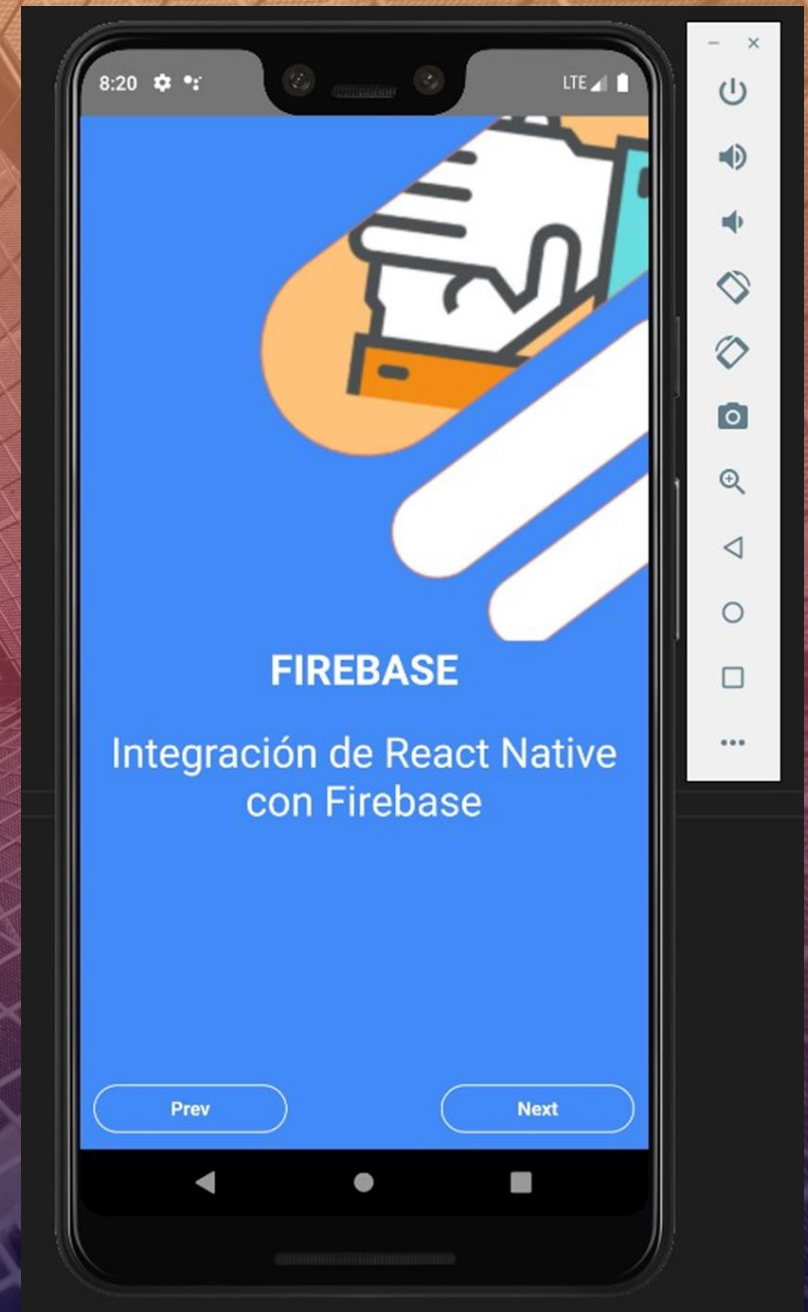
```
JS MainNavigator.js JS OnboardingFactory.js JS TaskScreen.js X JS Description.js JS But
src > Views > Onboarding > JS TaskScreen.js > [0] default
1  import React from 'react';
2  import { StyleSheet, View } from 'react-native';
3  import Constants from '../../Config/Constants';
4  import Images from '../../Config/Images';
5
6  import OnboardingFactory from '../../Components/Onboarding/OnboardingFactory';
7
8
9  const TaskScreen = ({navigation}) => {
10    return(
11      <OnboardingFactory
12        imagenes={Images.FDEFH3}
13        title1={Constants.STRING.DEF_H3}
14        des1={Constants.STRING.DESC2}
15        onPressn={()=>{navigation.navigate('AboutScreen')}}
16        onPressp={()=>{navigation.navigate('WelcomeScreen')}}
17      >
18      </OnboardingFactory>
19    );
20  };
21
22  export default TaskScreen;
```



Llamada de todos los componentes en el screen

AboutScreen

```
src > Views > Onboarding > JS AboutScreen.js > [X] default
1  import React from 'react';
2  import { StyleSheet, View } from 'react-native';
3  import Constants from '../../Config/Constants';
4  import Images from '../../Config/Images';
5
6  import OnboardingFactory from '../../Components/Onboarding/OnboardingFactory';
7  import Colors from '../../Config/Colors';
8
9
10 const AboutScreen = ({navigation}) => {
11   return(
12     <OnboardingFactory
13       imagenes={Images.FFIRE}
14       title1={Constants.STRING.FIRE}
15       des1={Constants.STRING.DESC3}
16       onPressn={()=>{navigation.navigate('Login')}}
17       onPressp={()=>{navigation.navigate('TaskScreen')}}
18     >
19     </OnboardingFactory>
20   );
21 };
22
23 export default AboutScreen;
```



Llamada de todos los componentes en el screen

Login a Firebase

```
src > Plugins > firebase > JS Firebase.js > ...
1  import firebase from 'firebase/app';
2  import 'firebase/firestore';
3  import 'firebase/auth';
4  import 'firebase/storage';
5  /*import {
6    API_KEY,
7    AUTH_DOMAIN,
8    DATABASE_URL,
9    PROJECT_ID,
10   STORAGE_BUCKET,
11   MESSAGING_SENDER_ID,
12   APP_ID
13 } from 'react-native-dotenv';*/
14
15 const firebaseConfig = {
16   apiKey: "AIzaSyC5shTWh8hHs6YtfJmCLleD_rTMHMx5Nyw",
17   authDomain: "defensah3pdm.firebaseio.com",
18   databaseURL: "https://defensah3pdm.firebaseio.com",
19   projectId: "defensah3pdm",
20   storageBucket: "defensah3pdm.appspot.com",
21   messagingSenderId: "961127200611",
22   appId: "1:961127200611:web:eaac4d1b609f69245c556"
23 };
24
25 const fire = firebase.initializeApp(firebaseConfig);
26 //export const firestore = firebase.firestore();
27 export default fire;
28
```



Firebase

Defensah3pdm

Ir a la documentación

Authentication

[Users](#) [Sign-in method](#) [Templates](#) [Usage](#)

Buscar por dirección de correo electrónico, número de tel... [Añadir usuario](#)

| Identificador | Proveedores | Fecha de creación | Inicio de sesión | UID de usuario ↑ |
|---------------------|-------------|-------------------|------------------|-------------------|
| djjomar28@gmail.com | | 10 ... | 10 ... | QMhZUaCk02N0... |
| dededed@gmial.com | | 6 m... | 6 m... | zs7YD8cCBrhBld... |

Filas por página: 50 1-2 de 2

Desarrollo

- Authentication
- Database
- Storage
- Hosting
- Functions
- ML Kit

Calidad

Crashlytics, Performance, Test Lab ...

Analytics

Dashboard, Events, Conversions, Au...

Crecimiento

Predictions, A/B Testing, Cloud Mes...

Conexión a Base
de datos