

## Examen de segundo parcial

Materia: aplicaciones web

maestro: Ray Brunet Parra Galaviz

La aplicación que se va a realizar  
va ser un login con sign in and sign up  
con Firebase.

"

```

import React & { useState, useEffect } from 'react';
import { View, Text, TextInput, Button, StyleSheet }
  from 'react-native';
import { initializeApp } from '@firebase/app';
import { getAuth, createUserWithEmailAndPassword,
  signInWithEmailAndPassword, onAuthStateChanged,
  signOut } from '@firebase/auth';

```

```

const FirebaseConfig = {

```

```

  apiKey: "AIzaSyj99QW15tWPxILcrXr",
  authDomain: "vital0323106539.firebaseio.com",
  projectId: "vital0323106539",
  storageBucket: "vital0323106539.appspot.com",
  messagingSenderId: "592244045",
  appId: "1:758759219409:web:each...",
  storageBucket: "gs://vital0323106539.appspot.com"
}

```



```
const app = initializeApp(firebaseConfig)
```

```
const AuthScreen = ({email, setEmail, password,  
  setPassword, isLogin, setIsLogin,  
  handleAuthentication}) => {
```

```
  return (
```

```
    <View style={styles.authContainer}>
```

```
      <Text style={styles.title}>Sign in / Sign up</Text>
```

```
      'Sign in' | 'Sign up' | 'Forgot Password'
```

```
      <Text input
```

```
        style={styles.input} value={email}
```

```
        value={password}
```

```
        onChangeText={setEmail} placeholder="password"
```

```
        secureTextEntry
```

```
      </Text input>
```

```
      <View style={styles.bottomContainer}>
```

```
        <Text style={styles.toggleText}>
```

```
          onpress={() => setIsLogin(!isLogin)}>
```

```
            {isLogin ? 'need an account' : 'already have an account'}
```

```
            Sign up | Sign in
```

```
          </Text>
```

```
        </View>
```

```
      </View>
```

```
    </View>
```

```
  )
```



```

const AuthenticatedScreen = ({ user,
  handleAuthentication }) => {
  return (
    <View style={styles.authcontainer}>
      <Text style={styles.title}>welcome</Text>
      <Text style={styles.emailText}>
        {user.email}</Text>
      <Button title="logout" onPress={
        handleAuthentication} color="#e74c3c">
        </Button>
    </View>
  );
};

```

```

export default App = () => {
  const [email, setEmail] = useState('');
  const [password, setPassword] = useState('');
  const [user, setUser] = useState(null);
  const [isLoading, setIsLoading] = useState(false);

```

```

const auth = getAuth(app);
useEffect(() => {
  const unsubscribe = onAuthStateChanged(
    auth, (user) => {
      setUser(user);
    }
  );

```

```

  return () => unsubscribe();
}, [auth]);

```



```
const handleAuthentication = async () =>
```

```
try {
```

```
  if (user) {
```

```
    console.log('user logged out successfully');
```

```
  } else {
```

```
    await create user with Email and
```

```
    password (auth, email, password);
```

```
    console.log('user created successfully');
```

```
  }
```

```
  catch (error) {
```

```
    console.log error('authentication error: ' +  
    errorMessage);
```

```
  }
```

```
};
```

```
return {
```

```
  <ScrollView contentContainerStyle={styles.  
    container}>
```

```
    {user ?
```

```
      <AuthenticationScreen user={user}>
```

```
        handleAuthentication={handleAuthentication}>
```

```
      </AuthenticationScreen>
```

```
    <AuthScreen
```

```
      email={email}
```

```
      setEmail={setEmail}
```

```
      password={password} setPassword={setPassword}
```

```
      isLogin={isLogin} </AuthScreen>
```



```
SetIsLogin = { SetIsLogin3  
handleAuthentication = { handleAuthentication  
1>
```

```
< }3
```

```
< /scrollview>
```

```
};
```

```
}
```