**Google Mail Login**

\* FireBase Console Create project

\*SHA certificate fingerprints Key past in project

\* open Gitbash terminal

\* cd android

\* cd app

\* keytool -list -v \ -alias androiddebugkey -keystore debug.keystore

\*password: android

\* SHA1 copy and past in firebase

\*Firebase console tap to open Authentication

\* Sign-in method Enable Google

\* Download google-services.json and past in andorid app folder

\* npm install to npm i @react-native-google-signin/google-signin

\*npm install --save @react-native-firebase/app

\*/android/build.gradle

buildscript {

dependencies {

*// ... other dependencies*

**classpath 'com.google.gms:google-services:4.3.14'**

*// Add me --- /\*

}}

\*/android/app/build.gradle

apply plugin: 'com.android.application'

**apply plugin: 'com.google.gms.google-services'** *// <- Add this line*

import {

  GoogleSignin,

  statusCodes,

} from '@react-native-google-signin/google-signin';

useEffect(() => {

    GoogleSignin.configure();

  }, []);

const googleLogin = async () => {

    try {

      await GoogleSignin.hasPlayServices();

      const userInfo = await GoogleSignin.signIn();

      console.log('User info', userInfo);

    } catch (error) {

      if (error.code === statusCodes.SIGN\_IN\_CANCELLED) {

        // user cancelled the login flow

        console.log(error);

      } else if (error.code === statusCodes.IN\_PROGRESS) {

        console.log(error);

        // operation (e.g. sign in) is in progress already

      } else if (error.code === statusCodes.PLAY\_SERVICES\_NOT\_AVAILABLE) {

        console.log(error);

        // play services not available or outdated

      } else {

        console.log(error);

        // some other error happened

      }

    }

  };

<TouchableOpacity style={styles.btnStyle} onPress={googleLogin}>

        <Text>Google Login</Text>

      </TouchableOpacity>

**Facbook Login**

\* Open facebook developer console

\*Open My App tab, open create App, click None and Next

\*App display name you can use facebook crate app name (eg: socialApp)

\* Click Product Facebook Login setup Android or ios

\* npm install to react-native-fbsdk-next

\*/android/build.gradle

repositories {

        google()

        mavenCentral() // <- Add this line

    }

\*/android/app/build.gradle

dependencies {

    implementation fileTree(dir: "libs", include: ["\*.jar"])

    implementation 'com.facebook.android:facebook-android-sdk:latest.release' // <- Add this line

\* Facebook developer console

\* Add package name (eg: your app name)

\* Default Activity Class Name (eg: your app name.MainActivity)

\* open Gitbash terminal

\* cd android

\* cd app

\* keytool -exportcert -alias androiddebugkey -keystore debug.keystore | openssl sha1 -binary | openssl base64

\*password: android

\* Copy the haskey and past facebook developer console Key Hashes

\* /app/res/value/string.xml

\*<resources>

    <string name="app\_name">googlefbauth</string>

    <string name="facebook\_app\_id">694364865312729</string>

    <string name="fb\_login\_protocol\_scheme">fb694364865312729</string>

    <string name="facebook\_client\_token">9f69e5ad4f230ad8d2b4a7220ac3aed5</string>

</resources>

\*./android/app/src/main

<uses-permission android:name="android.permission.INTERNET" />

<activity android:name="com.facebook.FacebookActivity"

        android:configChanges=

                "keyboard|keyboardHidden|screenLayout|screenSize|orientation"

        android:label="@string/app\_name" />

    <activity

        android:name="com.facebook.CustomTabActivity"

        android:exported="true">

        <intent-filter>

            <action android:name="android.intent.action.VIEW" />

            <category android:name="android.intent.category.DEFAULT" />

            <category android:name="android.intent.category.BROWSABLE" />

            <data android:scheme="@string/fb\_login\_protocol\_scheme" />

        </intent-filter>

    </activity>

import {

  LoginManager,

  GraphRequest,

  GraphRequestManager,

} from 'react-native-fbsdk-next';

const fbLogin = resCallback => {

    LoginManager.logOut();

    return LoginManager.logInWithPermissions(['email', 'public\_profile']).then(

      result => {

        console.log('result----', result);

        if (

          result.declinedPermissions &&

          result.declinedPermissions.includes('email')

        ) {

          resCallback({message: 'Email is required'});

        }

        if (result.isCancelled) {

          console.log('error');

        } else {

          const infoRequest = new GraphRequest(

            '/me?fileds=email,name,picture,friend',

            null,

            resCallback,

          );

          new GraphRequestManager().addRequest(infoRequest).start();

        }

      },

      function (error) {

        console.log('Login fail. with error:' + error);

      },

    );

  };

const onFbLogin = async () => {

    try {

      await fbLogin(\_responseInfoCallBack);

    } catch (error) {

      console.log('error raised', error);

    }

  };

  const \_responseInfoCallBack = async (error, result) => {

    if (error) {

      console.log('error tap', error);

      return;

    } else {

      const userData = result;

      console.log('fb data====', userData);

    }

  };

<TouchableOpacity

        style={{...styles.btnStyle, marginTop: 10}}

        onPress={onFbLogin}>

        <Text>FB Login</Text>

      </TouchableOpacity>