

# 海游登录SDK

## 一、基本信息

- SDK版本：1.1.1

### 海游登录SDK

- 一、基本信息
- 二、快速对接
  - 1. 准备工作
  - 2. 配置 AndroidManifest.xml
  - 3. build.gradle 配置
  - 4. 使用
- 三、用户参数说明
- 四、更新记录

## 二、快速对接

### 1. 准备工作

1. 解压 haiyou\_login\_1.1.zip，其中 HaiLoginDemo.zip 是demo
2. 复制 libs 中的文件到自己的项目 libs 中

### 2. 配置 AndroidManifest.xml

- 权限声明

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
```

- Application声明

```
<meta-data
    android:name="wallet_app_id"
    android:value="APP ID" /> 代码中的优先使用

<!-- 海游登录sdk 配置 -->
<activity
    android:name="com.walletfun.login.ui.RegisterOrForgotActivity"
    android:configChanges="screenSize|keyboardHidden|orientation" />

<activity
    android:name="com.walletfun.login.ui.LoginActivity"
```

```

        android:configChanges="screenSize|keyboardHidden|orientation" />

<!-- facebook 登录 -->
<meta-data
    android:name="com.facebook.sdk.ApplicationId"
    android:value="@string/facebook_app_id" />

<!-- 注意： 951132541753416 这个值请更换成 facebook_app_id 的值-->
<provider
    android:name="com.facebook.FacebookContentProvider"
    android:authorities="com.facebook.app.FacebookContentProvider951132541753416"
    android:exported="true" />

<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>

<!-- facebook end -->

```

### 3. build.gradle 配置

```

minSdkVersion 16 ( 大于等于16 )

// aar 必须配置的
repositories {
    flatDir {
        dirs 'libs'
    }
}

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])

    // 海游 SDK
    implementation(name: 'haiyou-login-2.0', ext: 'aar')
}

```

```

implementation(name: 'haiyou-common2.0', ext: 'aar')

// 其他第三方 SDK 有两种方式 根据自己开发环境选择一种
// 1 、如果支持网络库的依赖
implementation 'com.facebook.android:facebook-android-sdk:4.29.0'
implementation 'com.facebook.android:facebook-login:4.29.0'
implementation 'com.google.android.gms:play-services-auth:16.0.1'

// 2 、本地库依赖
implementation(name: 'play-services-auth-16.0.1', ext: 'aar')
implementation(name: 'play-services-auth-base-16.0.0', ext: 'aar')
implementation(name: 'play-services-base-16.0.1', ext: 'aar')
implementation(name: 'play-services-basement-16.0.1', ext: 'aar')
implementation(name: 'play-services-tasks-16.0.1', ext: 'aar')

implementation(name: 'facebook-core', ext: 'aar')
implementation(name: 'facebook-common', ext: 'aar')
implementation(name: 'facebook-login', ext: 'aar')
implementation(name: 'cardview-v7-28.0.0', ext: 'aar')
}

```

## 4. 使用

### (1) 初始化

```

// 如果在appId 传 null ,则使用 AndroidManifest.xml 中配置的 wallet_app_id 作为 appId

// 可以在 Application 中初始化
walletLoginSdk.init(context, "appId");

```

### (2) 启用登录

```

// 定义登录助手
private walletLoginHelper helper;

private void init(){
    // 初始化登录助手
    helper = walletLoginHelper.getInstance(activity);

    // 谷歌登录如果需要id, 请设置下面的代码
    helper.setGoogleServerClientId("googleServerClientId");

    // 设置登录回调方法
    helper.setLoginCallback(new walletLoginCallback() {
        @Override
        public void onResult(walletLoginResult result) {
            // 登录结果
            switch (result.getStatus()) {
                case 1:

```

```

        // 成功
        loginSuccess(result.getUser());
        break;
    case 2:
        // 用户取消了登录
        break;
    default:
        // 登录失败, 获取失败详情
        Object fail = result.getFailed();
        loginFailure(result.getPlatform(), fail);
        break;
    }
}

});

// 查询上次登录是否有效, 目前谷歌登录需要每次进来都登录
helper.hrtUser(new walletUserCallback() {
    @Override
    public void onUserResult(walletUser user) {

        // 如果为 null 则说明登录过期, 或者是谷歌登录

        if (user == null) {
            // 调用登录
            login();
        } else {
            // 可以不重新登录
            loginSuccess(user);
        }
    }
});
}

@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
    // 很重要的一步,
    if (helper.handlerActivityResult(requestCode, resultCode, data)) return;
    super.onActivityResult(requestCode, resultCode, data);
}

@Override
public void onConfigurationChanged(Configuration newConfig) {
    super.onConfigurationChanged(newConfig);
    if (helper != null) helper.onConfigurationChanged(newConfig);
}

@Override
protected void onDestroy() {
    super.onDestroy();
    if (helper != null)
        helper.destroy();
}

```

```

}

private void login(){
    // 使用默认登录方式
    helper.showLogin(walletLoginHelper.LOGIN_DEFAULT);

    // 以下为单独调用代码，自定义登录界面
    // facebook
    // helper.showLogin(walletLoginHelper.LOGIN_PLATFORM_FACEBOOK);

    // google
    // helper.showLogin(walletLoginHelper.LOGIN_PLATFORM_GOOGLE);

    // 海游
    // helper.showLogin(walletLoginHelper.LOGIN_PLATFORM_HAIYOU);

    // 游客
    // helper.showLogin(walletLoginHelper.LOGIN_PLATFORM_GUEST);
}

private void loginSuccess(walletUser user){
    // 登录成功的平台
    int platform = user.getPlatform();

    // 用户在此平台上的id
    String platformUserId = user.getLoginPlatformId();

    // 用户的 token
    String token = user.getAccessToken();

    // 用户的昵称，可能为空为null
    String nickName = user.getNickname();

    // 用户头像 可能为 null 为空
    String userIcon = user.getUserIcon();
}

private void loginFailure(int platform, Object err){

    switch (platform) {
        case walletLoginHelper.LOGIN_PLATFORM_FACEBOOK:
            // 使用 facebook 登录失败
            if (exception instanceof FacebookException) {
                String msg = err.toString();
            }

            break;
        case walletLoginHelper.LOGIN_PLATFORM_GOOGLE:
            if (exception instanceof ApiException) {

```

```

        // 谷歌登录失败
        int errorCode = (ApiException) exception).getStatusCode();
    }
    break;
case walletLoginHelper.REQUEST_LOGIN_HAIYOU:
case walletLoginHelper.LOGIN_PLATFORM_GUEST:

    if (exception instanceof Exception) {
        // 错误信息
        String msg = exception.toString();
    }
    break;
default:
    // 未知原因
    break;
}

}
}

```

### 三、用户参数说明

public class WalletUser {

```

// 登录的用户名，一般没有 用户名，谷歌登录的邮箱
private String username;
// 用户昵称
private String nickname;

// 登录平台上的 id (用户ID)
private String loginPlatformId;
// 平台id
private int platform;
/**
 * 0 - 保密、未公开
 * 1 - 男
 * 2 - 女
 */
private int gender;
// 昵称
private String address;
// 邮箱
private String email;
// 生日
private String birth;
// 手机号
private String mobilePhone;

// 对开发者公开
private String accessToken;
// 头像url
private String userIcon;

```

## 四、更新记录

版本	日期	说明	其他
2.0.0	2018.11.16	完善功能	
1.0.0	2018.10.22	基本功能	