

集成指南

- 本文档介绍 Android Studio 开发环境下基于 Maven 方式集成的步骤。
- 本文档适用 SDK 版本：2.0.0.0 及以后
- 本文默认读者已经具有基础的 Android 知识，以及项目工程结构如下：

```
Getui_SDK_Demo_AS_official/  
|- app/   (项目主模块)  
|   |- libs/ (第三方库)  
|   |- src/  (代码目录)  
|   |- build.gradle (模块级 gradle 文件)  
|- gradle/  
|- build.gradle (顶层 gradle 文件)  
|- settings.gradle  
| .....  
|
```

注：其中“.....”表示省略其他与本教程无关的内容，以下“.....”表示相同意义，不再重复说明。

重点提示

注意：

1、需要集成com.getui:gtc:3.3.1.0及以上版本。

1. 创建个推应用

请参考 [创建应用](#) 获取相应的 AppID 信息。该信息在之后的步骤配置中将会使用。

2. Maven 集成

2.1 配置 Maven 库地址

在项目根目录 `build.gradle` 文件的 `allprojects.repositories` 块中，添加个推 maven 库地址 `maven { url "https://mvn.getui.com/nexus/content/repositories/releases/" }`，如下所示：

```
buildscript {  
    repositories {  
        mavenCentral()  
        google()  
    }  
    dependencies {  
        .....  
    }  
}  
  
allprojects {
```

```

repositories {
    mavenCentral()
    google()
    maven { url 'https://jitpack.io' }
    //个推仓库地址
    maven {
        url "https://mvn.getui.com/nexus/content/repositories/releases/"
    }
}
}

```

2.2 配置依赖

1. 配置 SDK 依赖及应用参数：在 `app/build.gradle` 文件的 `dependencies` 块中引用个推 SDK 依赖 `implementation 'com.getui:aisdk:${version}'`，此处的 `${version}` 为对应的 SDK 版本号，并在 `android.defaultConfig` 下添加 `manifestPlaceholders`，配置个推相关的应用参数，如下所示：

```

.....

android {
    defaultConfig {
        manifestPlaceholders = [
            GETUI_APPID      : "your appid",
        ]
    }
    .....
}

dependencies {
    api 'com.getui:aisdk:2.0.0.0' //AI SDK
    api 'com.getui:gtc:3.3.1.0'  //个推核心组件
}

```

2.3 权限申明（重要）

为确保AISDK能稳定运行，SDK需要以下权限，aar包中的manifest已默认添加

```
</manifest>
```

```
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
```

```
</manifest>
```

3. 编写集成代码

3.1 初始化 SDK

SDK需要先初始化才能正常运行，建议您在Application或首页的OnCreate方法中，初始化AISDK。请参考下列代码实现初始化方法：

```
AiManager.getInstance().initialize(getApplicationContext(), new Callback<String>() {
    @Override
    public void onFailure(Throwable t) {
        //初始化失败
        Log.e("AISDK", "初始化失败", t);
    }

    @Override
    public void onResponse(String response) {
        //初始化成功，返回 'sdk init success'
        Log.d("AISDK", "初始化成功: " + response);
    }
});
```

注意：后续SDK所有的API调用都依赖初始化成功，请确保 `onResponse` 完成后再调用其他接口

3.2 加载智能体页面

SDK需要确保初始化成功后才能进行AI会话

```
//true表示打开智能体页面；false表示不打开，开发者可根据response里的url自行处理
AiManager.getInstance().loadAgent("{替换为你的agentId}", true, new Callback<String>() {
    @Override
    public void onFailure(Throwable t) {
        Log.e("AISDK", "loadAgent failed", t);
    }

    @Override
    public void onResponse(String response) {
        Log.e("AISDK", "loadAgent success");
    }
});
```

成功拉起聊天页如下



◆ 有没有种田文推荐



◆ 有趣又带点深度的东京流小说

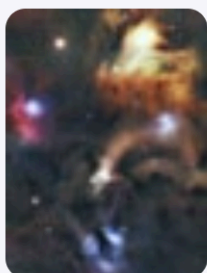


◆ 母鸡卡元素的小说



科幻小说推荐

探索未知世界，开启
科幻之旅



三体

刘慈欣

科幻

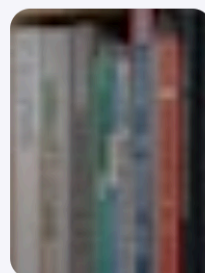
硬核

★★★★★ 9.4
分



文学经典推荐

品味人生百态，感
受文学魅力



活着

余华

文学

现实

★★★★☆ 9.2
分

有什么需要问我的吗~



以上内容由AI生成，请谨慎甄别。

15:37



5G 5G 53%

🗑️ 清空

🔊 播放

查看更多科幻小说

数据更新于：2024-01-15

^

查看更多文学经典

数据更新于：2024-01-15

15:37

请推荐更多科幻类小说

🔗 思考完成 ^

好，我现在需要帮用户推荐更多科幻类的小说。用户特别提到了“科幻类”，所以我要根据知识库里的书籍信息，筛选出符合科幻元素的小说。

首先，我要理解用户的需求，科幻类小说通常包括未来科技、外星文明、时空穿梭、机甲战斗等元素。知识库里有10本书，我需要逐一查看每本书的简介、标签和内容，看看哪些符合科幻主题。

第一本书是《我们来自未来方舟！》，标签里

有“穿越、修罗场同人、战斗、治愈”，看起来涉及时空穿梭和科幻元素，符合要求。

第二本是《赛博朋克之未来神话》，明确提到赛博朋克、灵能、AI，这些都是典型的科幻元素，应该推荐。

第二本《很对劲的言谈智趣品兼时空管理局临

有什么需要问我的吗~



以上内容由AI生成，请谨慎甄别。