

容联·七陌V7访客端 SDK Android v1.5.0文档

一、概述

[sdk环境和依赖库](#)

[sdk Demo页面](#)

二、SDK功能接口

[sdk导入与集成](#)

[具体步骤](#)

[权限说明](#)

[sdk](#)

[Demo](#)

[sdk初始化配置与回调](#)

[1：初始化sdk需要的帮助类](#)

[2：初始化接入客服配置](#)

[3：进行初始化客服](#)

[4：初始化回调](#)

[聊天页面跳转](#)

三、高级功能配置

[混淆](#)

[创建新消息发送](#)

[1：创建消息](#)

[2：将消息插入数据库](#)

[3：发送消息](#)

[获取当前会话坐席接入状态](#)

[转人工接口](#)

[自定义样式与功能](#)

[自定义订单卡片](#)

[自定义列表中横向滑动快捷按钮](#)

[消息列表数据点击事件](#)

[富文本点击事件](#)

[Log文件查看](#)

[查看SDK版本号](#)

四、常见问题

[1：依赖库冲突问题](#)

[2：so库支持类型](#)

五、升级日志

[V1.5.0 \(2022-02-25\)](#)

[V1.4.1 \(2021-12-08\)](#)

[V1.4.0 \(2021-11-23\)](#)

一、概述

容联·七陌V7 平台安卓SDK，可由对接企业移动端开发人员通过集成开发实现App内访客端的接入。

我们提供了较为完善的Demo以及Demo源码。对接之前，对接企业的负责人或已开通坐席平台的管理员可以下载Demo提供给研发进行运行体验，提前了解SDK能力便于更快速的进行SDK集成和开发工作。

目前的UI属性提供部分自定义配置，可在管理端进行配置，如您需要对Demo ui样式进行更多的定制开发请详细了解Demo代码后进行自定义ui。

sdk环境和依赖库

环境版本

- minSdkVersion： 19
- targetSdkVersion： 29
- gradle插件版本： 最低 build:gradle:3.0.0
- gradle版本： 最低 gradle-4.1

依赖库

demo

– `com.github.bumptech.glide:glide:4.9.0` (可选)

sdk

– `com.github.7moor-tech:7moor_baseLib_sdk`

```

|-org.greenrobot:eventbus:3.2.0
|-com.google.code.gson:gson:2.8.5
|-com.qiniu:qiniu-android-sdk:8.4.0
|-com.j256.ormlite:ormlite-core:5.1
|-com.j256.ormlite:ormlite-android:5.1

```

sdk Demo页面

- 会话页面：MoorChatFragment.java
- 满意度评价弹窗：MoorEvaluationDialog.java
- 会话内容类型item：com.moor.imkf.Demo.multichat.multirow包下

具体消息类型如下：

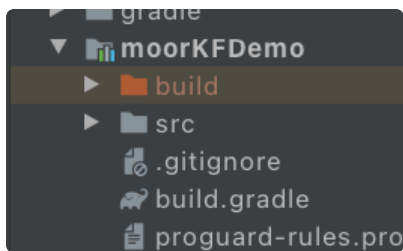
MoorCSRRecevedViewBinder	满意度评价类型消息
MoorFastBtnReceivedViewBinder	横向快速点击按钮类型消息
MoorFileReceivedViewBinder MoorFileSendViewBinder	附件收发类型消息
MoorFlowListMultiSelectViewBinder	机器人数据多选类型消息
MoorFlowListSingleHorizontalViewBinder	机器人数据单选类型横向滑动消息
MoorFlowListSingleVerticalViewBinder	机器人数据单选类型竖向滑动消息
MoorFlowListTextViewBinder	机器人数据单选类型竖向文字列表消息
MoorFlowListTwoViewBinder	机器人数据两列滑动按钮
MoorImageReceivedViewBinder MoorImageSendViewBinder	图片收发类型消息

MoorLogisticsRecevedViewBinder	机器人类型物流节点数据类型消息
MoorOrderListSendViewBinder	发送订单卡片类型消息
MoorRobotUseFulReceivedViewBinder	机器人有无帮助类型消息
MoorSystemReceivedViewBinder	系统提示类型消息
MoorTextReceivedViewBinder MoorTextSendViewBinder	文本收发类型消息
MoorVoiceSendViewBinder	语音类型消息
MoorXbotTabQuestionViewBinder	机器人常见问题分组类型消息

二、SDK功能接口

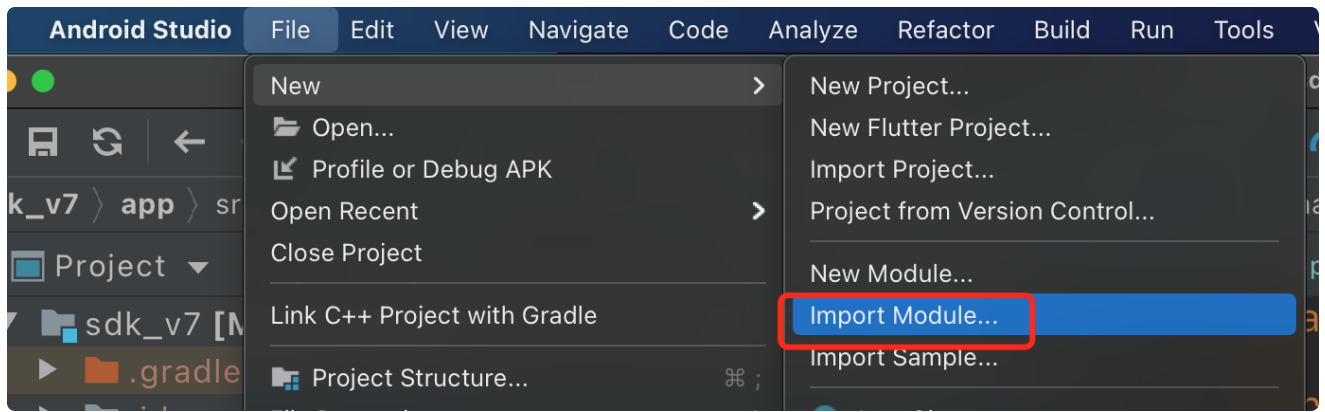
sdk导入与集成

将Demo中的moorKFDemo模块作为Module引入自己的项目中

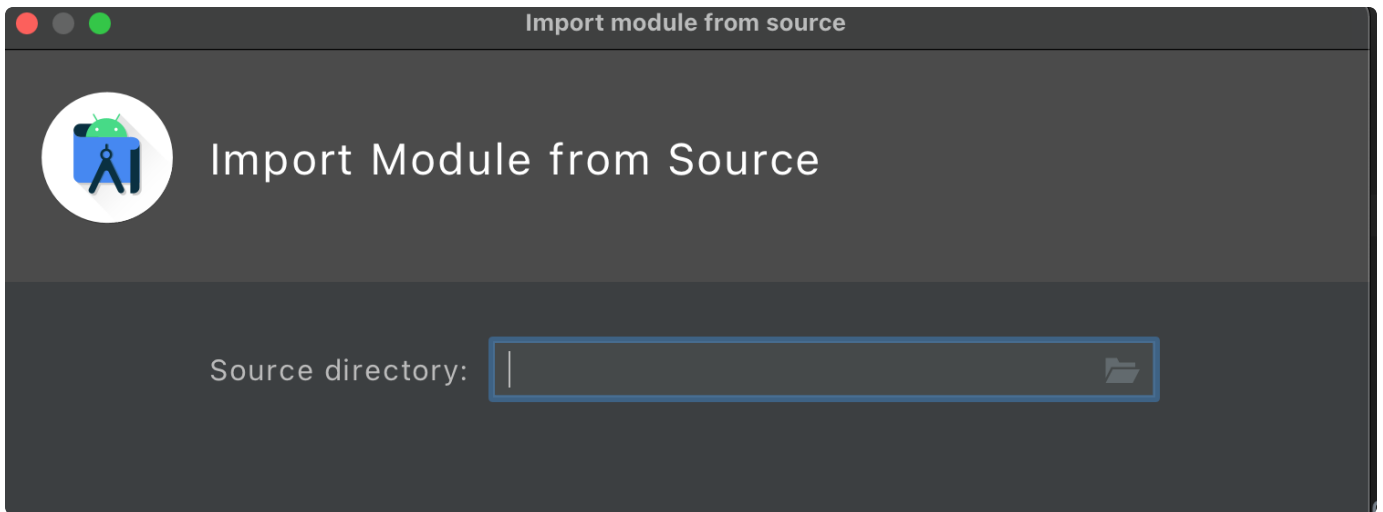


具体步骤

1. 使用Android Studio 导入Module



2. 窗口中选择moorKFDemo模块进行下一步导入



3. 在自己项目所需模块，如app目录下的build.gradle中添加依赖

```
implementation project(path: ':moorKFDemo')
```

4. 重新编译项目

备注：

- 如无特定场景需求，本Demo示例包含大部分需求，用户将moorKFDemo导入到自己项目后，填入必填项即可完成配置。
- sdk初始化配置可参考下文中：[初始化配置](#)
- 如有样式配置可参考下文中：[高级功能配置](#)

权限说明

sdk

- android.permission.INTERNET
sdk需要使用到网络
- android.permission.ACCESS_NETWORK_STATE

sdk内部需监听网络状态变化，用于会话长连接保持

- android.permission.WRITE/READ_EXTERNAL_STORAGE
文件存储读写，sdk需要对会话数据持久化存储以及必要的数据存储

Demo

- android.permission.CAMERA
照相机权限，Demo提供拍照发送图片功能，如不需要可删除
注意：删除此权限 要隐藏对应的功能
- android.permission.RECORD_AUDIO
录音机权限，Demo提供拍照发送语音功能，如不需要可删除
注意：删除此权限 要隐藏对应的功能

sdk初始化配置与回调

1：初始化sdk需要的帮助类

参考demo中的DemoAPP Application类文件

在您的app Application onCreate()方法中添加以下代码初始化所需资源

```
1 public class DemoApp extends Application {
2     @Override
3     public void onCreate() {
4         super.onCreate();
5         //初始化 MoorMoorUtils
6         MoorUtils.init(this);
7     }
8 }
9 }
```

Plain Text

复制代码

2：初始化接入客服配置

参考Demo中 MainActivity 中初始化配置；

必填项

- accessId：从PC座席端获取

- userId: 访客id (自定义)
- userName: 访客昵称 (自定义)
- BaseImageLoader: sdk中图片加载器, Demo中以Glide为例, 可参考 `GlideImageLoader` 实现自己的加载器

Java | 复制代码

```

1    //UI配置参数
2        MoorOptions options = setMoorOptions();
3        MoorImageLoader loader = setMoorImageLoader();
4        //配置SDK所需内容以及自定义配置
5        MoorConfiguration configuration = new
MoorConfiguration.MoorBuilder()
6            //必填 设置初始化SDK所需 accessId
7            .setAccessId('')
8            //必填 设置初始化访客的id
9            .setUserId('')
10           //必填 设置初始化访客的昵称
11           .setUserName('')
12           //选填 设置访客头像
13           .setUserHeadImg('')
14           //必填 sdk的服务地址类型
15           .setServiceType(MoorEnumServiceType.T_REQUEST)
16           //必填 设置图片加载器
17           .setBaseImageLoader(loader)
18           //选配 设置sdk中一些错误code是否Toast提示 默认true
19           .isShowSdkToast(true)
20           //选配 设置是否打印Log 默认true
21           .isLogOpen(true)
22           //选配 设置是否将日志输出到文件 默认true
23           .isLog2File(true)
24           .build();
25

```

注意事项

1. Demo提供默认的以Glide实现的图片加载器GlideImageLoader, 如用户自己项目中图片加载使用的其他框架, 可参考此类, 实现IMoorImageLoader接口即可;
2. 具体UI配置参考下文: **自定义样式与功能**

3: 进行初始化客服

初始化SDK

`MoorOpenChatHelper.getInstance().initSdk(configuration);`

4: 初始化回调

(可选)参考 `MoorOpenChatHelper` 中 `MoorOpenChatListener` 实现示例

Java | 复制代码

```
1  IMoorInitListener
2      /**
3       * 初始化回调开始
4       */
5      void onInitStart();
6
7      /**
8       * 开启服务
9       */
10     void onStartService();
11
12     /**
13      * 初始化回调成功
14      *
15      * @param msg
16      */
17     void onInitSuccess(String msg);
18
19     /**
```

聊天页面跳转

需等待 `IMoorInitListener` 回调 `onInitSuccess()` 后执行跳转

参考Demo中 `MoorOpenChatHelper.openChat()` 方法

Java | 复制代码

```
1      /**
2       * 开启会话页面
3       */
4      public void openChat() {
5          Intent intent = new Intent();
6          intent.setClass(MoorActivityHolder.requireCurrentActivity(),
7              MoorChatActivity.class);
8          MoorActivityHolder.requireCurrentActivity().startActivity(intent);
9      }
```


三、高级功能配置

混淆

如果你的项目编译开启混淆

请在app proguard 配置文件中加入以下代码：

```
1  #moorsdkdemo
2  -dontwarn com.moor.imkf.demo
3  -keep class com.moor.imkf.demo.** {*; }
4
5  #Glide(可选)
6  -keep public class * implements com.bumptech.glide.module.GlideModule
7  -keep class * extends com.bumptech.glide.module.AppGlideModule {
8      <init>(...);
9  }
10 -keep public enum com.bumptech.glide.load.ImageHeaderParser$** {
11     **[] $VALUES;
12     public *;
13 }
14 -keep class
    com.bumptech.glide.load.data.ParcelFileDescriptorRewinder$InternalRewinder {
15     *** rewind();
16 }
```

创建新消息发送

1：创建消息

Demo提供了MoorMsgHelper帮助类可以提供多种类型消息的快速创建。

2：将消息插入数据库

MoorMsgDao.getInstance().addMsg(msgBean);

MoorChatFragment中提供的sendMsgToPage()方法中有调用示例，并将消息插入到数据源List同时移动到底部功能。

3：发送消息

使用MoorMsgHelper构造了消息后通过JobManager管理类进行消息发送。

MoorChatFragment中提供的jobManagerSendMsg(MoorMsgBean)方法进行消息的发送

Java | 复制代码

```
1      /**
2       * 消息队列发送消息
3       *
4       * @param msgBean
5       */
6      private void jobManagerSendMsg(MoorMsgBean msgBean) {
7          JobManager jobManager = MoorUtils.getJobManager();
8          MoorMessageJob messageJob = new MoorMessageJob(msgBean);
9          jobManager.addJobInBackground(messageJob);
10     }
```

获取当前会话坐席接入状态

通过MoorManager.getInstance().getChatStatus()可以获取当前会话的接入状态

```
/**
 * 当前是排队状态
 */
MoorConstants. NOW_QUEUE_STATUS

/**
 * 当前人工状态
 */
MoorConstants. NOW_CLAIM_STATUS

/**
 * 当前机器人状态
 */
MoorConstants. NOW_ROBOT_STATUS

/**
 * 当前初始化状态
 */
MoorConstants. NOW_INIT_STATUS
```

/**

* 当前是否进入留言状态

*/

MoorConstants. *MOOR_LEAVETAG_STATUS*

转人工接口

MoorChatFragment 中提供的convertManual()方法可以进行转人工操作。

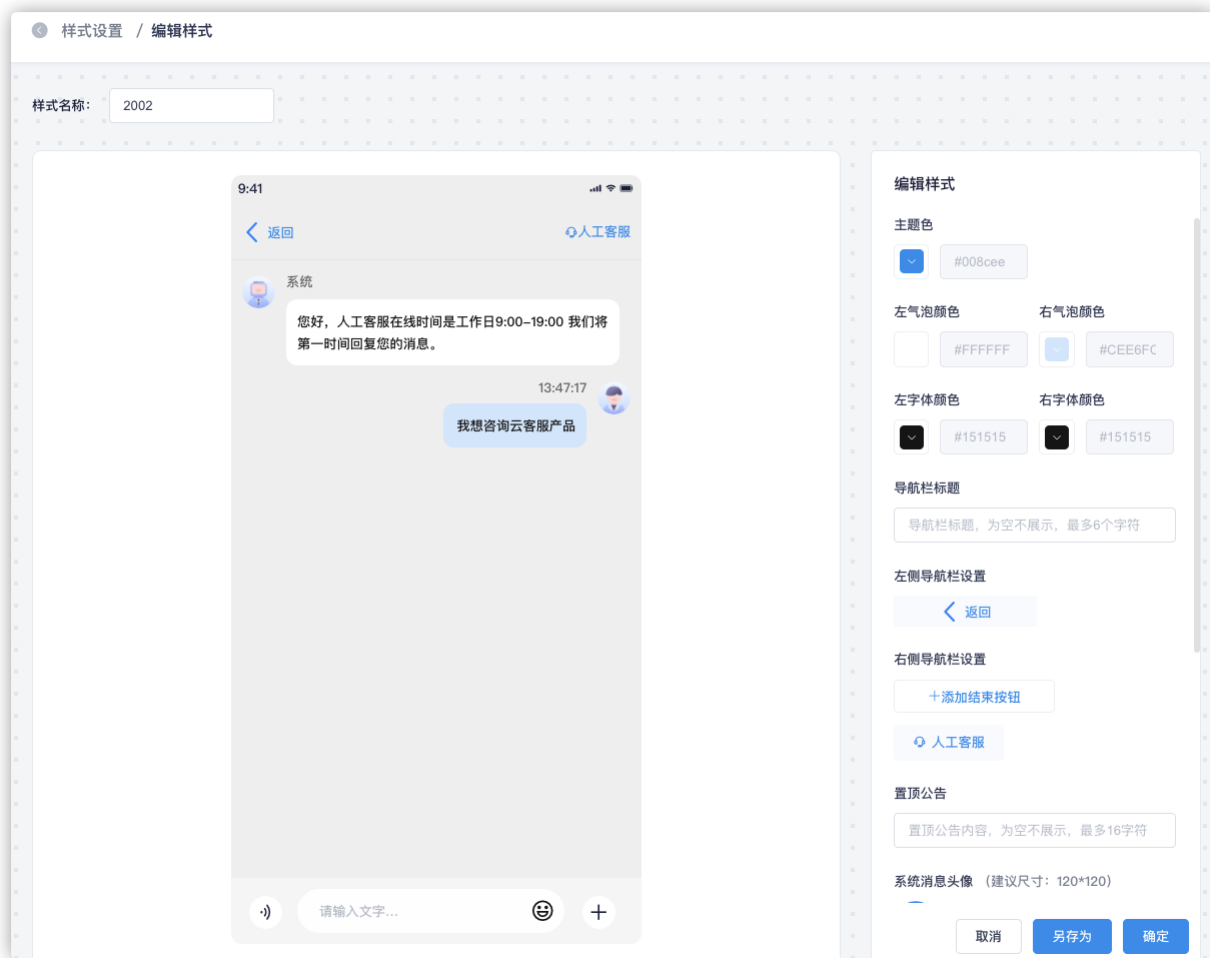
参数：keyManual (转人工关键词，一般为消息中配置，如果自己实现转人工按钮操作的话参数传空即可)。

注意：转人工时间只能在当前为机器人状态下触发

自定义样式与功能

SDK提供网络动态获取主题色以及消息气泡颜色等方法

可登录坐席端SDK渠道设置进行操作修改



SDK本地对应UI配置接收实体：MoorOptions

MoorChatFragment onCreateView中已经获取网络拉取保存在数据库中的MoorOptions对象，进行ui的渲染。

注意：网络配置的ui优先级高于本地xml配置的颜色

如需对颜色或样式进行本地修改则需要在获取MoorOptions后通过提供的get、set方法进行修改，同时使用

MoorOptionsDao.createOrUpdate()进行数据更新。

如过需要在机器人咨询模式下可隐藏加号菜单的按钮则在拉取到的MoorOptions对象对数据进行添加

如：MoorChatFragment中示例的

```

@Override
public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {
    moorChatView = View.inflate(mActivity, R.layout.moor_activity_chat, root: null);
    chatHandler = new ChatHandler( fragment: this);
    MoorManager.getInstance().setSDKInitFinish(true);
    LoadingDialog = MoorLoadingDialog.create();

    preLoaderId = getArguments().getInt(MoorDemoConstants.MOOR_PRE_LOADER_ID, defaultValue: 0);

    options = MoorOptionsDao.getInstance().queryOptions();

    //机器人状态下是否显示加号，默认true展示
    options.setRobotShowPanelButton(true);

    MoorManager.getInstance().setOptions(options);
}

```

自定义订单卡片

功能介绍：

可以发送给坐席端包含订单信息的卡片式消息，内部可以嵌入图片，标题，订单号，以及点击事件等功能

注： 机器人状态下可以发送但是暂时不能响应订单卡片的内容，当人工接入后会在消息历史中看到卡片消息



使用：

调用 `MoorUtils.createOrderCardInfoMsg()` ，发送订单卡片

```

1  /**
2   * 创建发送订单卡片数据
3   *
4   * @param orderTitle    订单标题 必填
5   * @param orderNumName  订单号号称
6   * @param orderNum      订单编号 必填
7   * @param btnRightShow  右侧按钮是否展示
8   * @param btnRightText  右侧按钮文字
9   * @param btnRightTarget 右侧按钮点击类型
10  * @param btnRightUrl   右侧按钮点击数据
11  * @param btnLeftShow   左侧按钮是否展示
12  * @param btnLeftText   左侧按钮文字
13  * @param btnLeftTarget 左侧按钮点击类型
14  * @param btnLeftUrl    左侧按钮点击数据
15  * @param content       描述数据
16  * @param price         价格
17  * @param imgUrl        图片

```

可参考 `MoorChatFragment.sendOrderCard()` 示例发送订单卡片

自定义列表中横向滑动快捷按钮

功能介绍：

在进线前配置本数据可以在sdk聊天页面进入后的消息尾部增加横向滑动的卡片式按钮

- 按钮整体会跟随消息列表上下滑动
- 按钮仅在sdk初始化进入后加载一次，退出后会移除，不会保存在历史消息中
- 按钮默认点击事件为发送数据中的clickText文字



使用：

初始化配置对象

```
MoorConfiguration configuration = new MoorConfiguration.MoorBuilder()
```

....

```
//添加消息列表中横向滑动快速按钮数据
```

```
.setFastBtnBeans(getFastBtn())
```

.build();

getFastBtn()创建数据方法，在Demo中MainActivity中提供演示
字段说明：

imgUrl	图片地址
showName	展示的文字
clickText	点击要发送的文字
bgColor	每个的背景颜色

消息列表数据点击事件

列表adapter中点击事件封装于 MoorChatFragment 中；

具体类型： MoorEnumChatItemClickType

Java | 复制代码

```
1      /**
2      * adapter中的点击和长按等事件
3      */
4      private final IMoorBinderClickListener adapterClick = new
      IMoorBinderClickListener() {
5
6          @Override
7          public void onClick(View v, final MoorMsgBean item,
      MoorEnumChatItemClickType type) {
8              }
9
10         @Override
11         public void onLongClick(View v, MoorMsgBean item, Object o) {
12
13         }
14     };
```

富文本点击事件

MoorTextParseUtil类为富文本解析管理类

HttpClickSpan默认提供了跳转系统浏览器的方法

AClickApan提供机器人配置转人工和电话号码的点击事件

NumClickSpan 提供了数字正则点击拨号事件

Log文件查看

本地日志文件默认路径：

/storage/emulated/0/Android/data/com.moor.imkf.sample/files/Download/Moor_Log/

查看SDK版本号

查看当前sdk版本号

String version= MoorConstants.MOOR_SDK_VERSION

四、常见问题

1：依赖库冲突问题

如果遇到gson或eventbus...等依赖库冲突可以使用gradle提供的依赖排除方法对SDK提供的依赖进行排除，引用您项目的依赖库。

以gson举例：

(xxxxx){exclude group:'com.google.code.gson'}

2：so库支持类型

so库支持：

'armeabi'

'armeabi-v7a'

'arm64-v8a'

'x86'

'x86_64'

配置方式：

在moorKFDemo和app 层的defaultConfig 添加对应所需要的abiFilters即可

```
defaultConfig { DefaultConfig it ->
    minSdkVersion gradle.ext.minSdkVersion
    targetSdkVersion gradle.ext.targetSdkVersion
    versionCode gradle.ext.versionCode
    versionName gradle.ext.versionName
    multiDexEnabled true
    consumerProguardFiles 'moor-rules.pro'
    ndk { NdkOptions it ->
        // 设置支持的SO库架构
        abiFilters 'arm64-v8a', 'armeabi-v7a'
    }
}
```

五、升级日志

V1.5.0 (2022-02-25)

- 调整依赖库结构
- 增加访客黑名单功能
- sdk内部优化

V1.4.1 (2021-12-08)

- Android 增加release编译后 文件日志输出加密

V1.4.0 (2021-11-23)

- 消息已读未读状态显示
- 访客输入内容坐席预知
- 支持坐席撤回消息状态显示
- 历史消息排序与加载优化
- sdk体验和页面展示性能优化

