

一、简介

1、IVR 简介

IVR(Interactive Voice Response)即互动式语音应答，是一种功能强大的电话自动服务系统。它用预先录制或TTS文本转语音技术合成的语音进行自动应答的系统，提供一种为客户进行菜单导航的功能。您只须用电话即可进入服务中心，可以根据操作提示收听相关产品信息，也可以根据用户输入的内容播放有关的信息。

云通讯平台IVR有两种接入方式：

- (1)外呼方式：通过调用云通讯平台提供的rest IVR外呼接口；
- (2)呼入方式：通过落地电话呼入云通讯平台提供的IVR接入号或者通过云通讯平台提供的VOIP呼入serviceid。

2、集成流程

- (1)注册成为云通讯平台用户。
- (2)登陆云通讯平台，进入管理控制台，创建自己的应用。
- (3)下载IVR demo布置到服务器后进行测试体验。

3、前期准备

- (1)注册成为云通讯的用户，详见 [新手指引](#)
- (2)登陆云通讯平台，进入管理控制台，创建自己的应用，IVR只能用创建应用进行测试，demo应用不支持IVR测试。在创建IVR类应用的时候，需填写应用名称和选择应用对应的行业，并且勾选"启用IVR"后在 "IVR通知消息URL" 中填写服务器的回调地址，如果需要用的文本转语音的业务就需要勾选"启用TTS"，不必选择"启用应用回调地址"、"服务器白名单"等选项。

IVR 通知消息 URL 填写注意：

1).URL 中如果没有加上端口则是默认端口 80；

2).云通讯平台只支持 http 的消息发送，如果填写 https 则无法回调；

The screenshot shows the '创建应用' (Create Application) page in the Yuntongxun console. The left sidebar has a blue header with 'Applications' and a menu with '应用' (Applications) highlighted. The main content area has a form with the following fields and options:

- * 应用名称: myivr (不超过20个字符，请按照应用审核规范)
- 启用回调地址: ☐ 勾选启用
- 启用IVR: ☒ 勾选启用 (收费功能)
- IVR通知消息URL: http://www.yuntongxun.com/ivr (基础通知消息、会议通知消息)
- 坐席管理功能开关: ☐ 勾选启用 (坐席通知消息)
- 启用TTS: ☐ 勾选启用 (收费功能)
- 服务器白名单: ☐ 勾选启用
- * 应用行业: 电子商务

Buttons at the bottom: 确定 (orange), 取消 (grey).

(3)应用创建成功后，在"应用列表"里面可以看见自己刚刚创建的应用，可以对该应用进行编辑和删除操作。当该应用集成完云通讯IVR能力后准备正式对外发布时，需要点击"上线"申请，此时应用才可以在云通讯的生产环境使用，否则只能在沙盒环境使用。

放音文件的格式样本如下：位速 128kbps，音频采样大小16位，频道 1(单声道)，音频采样级别 8 kHz，音频格式 PCM，这样能保证放音的清晰度。

4、核心概念

(1)生产环境：应用上线后的正式生产环境，对应的接入域名为：

<https://app.cloopen.com:8883>

(2)沙盒环境：应用开发测试时使用的调试环境，对应的接入域名为：

<https://sandboxapp.cloopen.com:8883>

(3)开发者主账号：每个开发者在云通讯平台对应一个主账号

(a)ACCOUNT SID：主账号id。

(b)AUTH TOKEN：主账号验证token。

(4)AppID：应用标识，每个创建的应用都对应唯一的id标识。

(5)IVR接入号：IVR呼入的号码，当应用上线后勾选“启用IVR”后由云通讯平台进行分配，一个接入号只能绑定一个应用，但是一个应用可以绑定多个接入号。需要在测试环境调试IVR呼入可以使用云通讯平台提供IVR测试接入号01082167088进行调试。

(6)serviceid：应用服务id，创建应用生成的绑定唯一id，用户VOIP的IVR接入号，等同于IVR接入号的作用，只能是绑定应用下面的子账号使用。暂时没有在官方网站提供，需要联系技术支持人员进行提供。

说明：每一个注册开发者对应一个主账号。

每一个主账号下可以创建多个App。

5、开发准备

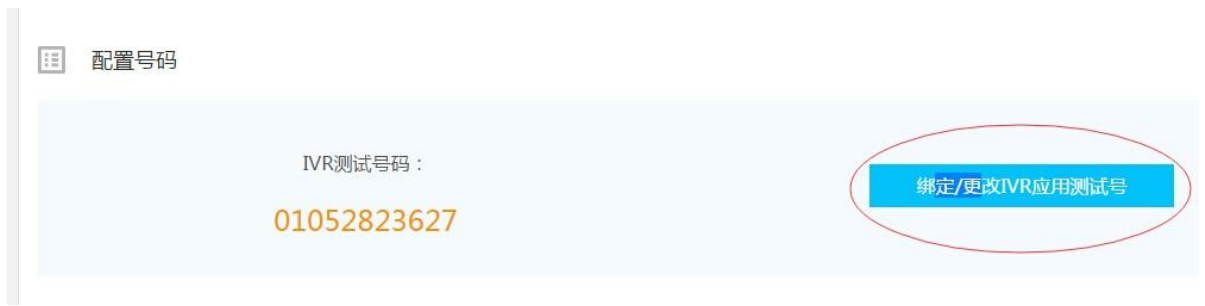
- (1) 解压缩有callcenterRest和callcenterIVR两个工程，menu.wav、timeout.wav、timeoutbye.wav、waitmusic.wav、busy.wav、bye.wav、pingjia.wav、thank.wav八个放音文件，以及一个IVR Demo开发文档；
- (2) callcenterIVR 目录下面 src 文件夹为工程代码，WebRoot 文件夹下面是做服务端布置到自己的服务器环境中，如果布置在根目录下面，在创建应用的时候“IVR 通知消息 URL”请填写 http://ip:port/ ，如果是在根目录的目录下面，例如根目录的 callcenterIVR 目录下面，“IVR 通知消息 URL”请填写 http://ip:port/callcenterIVR/ ，如果需要用域名就把 ip 更换为域名；
- (3)把解压的八个放音文件通过云通讯的官网上传进行审核；
- (4) callcenterRest 工程需要做为客户端布置到服务器上，通过 createqueue 进行创建队列，通过 agentonwork 进行坐席的上班，通过 agentready 进行坐席准备就绪；
- (5)确定 IVR 使用的是外呼方式还是呼入方式：
 - a．外呼方式请使用 IvrDial.java 是做为客户端用来发起 rest 请求消息；
 - b．呼入方式则不需要 IvrDial，通过呼叫 IVR 测试号码 01082167088 即可，因为所有的用户都是用 01082167088 的 IVR 接入号进行调试，所以需要在创建的应用下面进行绑定主叫号码，绑定完成后呼入就会路由到对应的应用，否则呼入只能是云通讯平台的体验。需要注意的是，多个应用下面不能绑定相同的主叫号码，否则只有一个有效。绑定主叫的号码步骤：
 - 1).在“应用”列表中“号码管理”的“测试号码”页面中填写测试的号码，然后进行验证通过；



2).在“应用”列表中“应用管理”的“应用列表”页面中点击需要绑定主叫号码的应用名称；



3).在应用名称页面中找到“配置号码”后点击“IVR 测试号码”后面的“绑定/更改 IVR 应用测试号”，在弹出的页面中勾选需要绑定的主叫号码，完成后点击确定就完成配置。



二、开发指南

1.Demo 的 IVR 流程

- 1.通过创建队列接口创建 1 和 2 队列；
- 2.通过坐席上班接口把坐席上班准备中，然后通过坐席准备就绪接口把坐席状态置为准备就绪；
- 3.用户呼入或者 IVR 外呼用户接听后放音提示按键，使用按键命令嵌套放音命令，放音循环播放并且设置按键超时时间，放音提示：欢迎致电，中文服务请按 1，英文服务请按 2；
- 4.用户按键 1 进入 1 队列转接 1 队列中的坐席，用户按键 2 进入 2 队列转接 2 队列中的坐席，设置进入排队超时时间。
- 5.增加进入排队的副命令调整，在用户等待中没有坐席接听的时候调整到副命令，然后放音提示：坐席正忙，继续等待请按 1，结束请挂机。
- 6.用户按 1 继续进入排队，按其他键直接挂机。
- 7.当坐席接听并且坐席挂机后通过回调放音提示用户：请对我们的客服人员做评价，满意请按 1，不满意请按 2；
- 8.用户超时没按键挂机，用户按 1 或者 2 后提示用户：感谢您的来电，再见！挂机

2.Demo 接口说明

(1) callcenterRest工程

RestSDK.java

发送 rest 座席上班接口

```
public static String agentonwork(String number,String agentid,String agenttype,String agentstate)
```

发送 rest 座席准备就绪接口

```
public static String agentready(String agentid,String action,String state,String priority,String force)
```

发送 rest 创建队列接口

```
public static String createqueue(String queueType,String typedes,String worktime,String offworkprompt,String offworkdate,String offworkweekday)
```

发送 rest IVR 外呼接口

```
public static String Dial(String number,String userdata,String record)
```

发送 https 请求消息接口

```
public static String sendPost(String url, String param)
```

(6) ivr.php

a.获取云通讯平台 IVR 通知消息 url 接口

```
String uri = req.getRequestURI();
```

b.获取云通讯平台 IVR 通知消息 url 中参数的接口

```
String appid = req.getParameter("appid");
```

c.IVR 启动业务接口

```
public String startservice(HttpServletRequest req)
```

响应包体

```
<?xml version='1.0' encoding='UTF-8'?>
  <Response>
    <Get action='dtmfreport' numdigits='1' timeout='20'>
      <Play loop='-1'>menu.wav</Play>
    </Get>
    <Play>timeout.wav</Play>
    <Redirect>gettimeout</Redirect>
  </Response>
```

说明：主命令为按键命令并且嵌套 menu.wav 的放音提示，在用户按键一个就回调通知到 dtmfreport 的地址，如果用户没有按键就会 20 秒超时后执行副命令，播放 timeout.wav 的提示放音后跳转通知。

d.IVR 用户按键超时接口

```
public String gettimeout(HttpServletRequest req)
```

响应包体

```
<?xml version='1.0' encoding='UTF-8' ?>
  <Response>
    <Get action='dtmfreport' numdigits='1' timeout='20'>
      <Play loop='-1'>menu.wav</Play>
    </Get>
    <Play>timeoutbye.wav</Play>
    <Hangup/>
  </Response>
```

说明：主命令为按键命令并且嵌套 menu.wav 的放音提示，在用户按键一个就回调通知到 dtmfreport 的地址，如果用户没有按键就会 20 秒超时后执行副命令，播放 timeoutbye.wav 的提示放音后挂机。

e.IVR 获取用户按键接口

```
public String dtmfreport(HttpServletRequest req)
```

如果用户按 1 的响应包体

```
<?xml version='1.0' encoding='UTF-8' ?>
  <Response>
    <EnterCCS queuetype='1' timeout='30' queuetimes='1'
    agenthangupurl='agenthangup'>
      <Play>waitmusic.wav</Play>
    </EnterCCS>
    <Redirect tag='1'>transferAgentFail</Redirect>
  </Response>
```

如果用户按 2 的响应包体

```
<?xml version='1.0' encoding='UTF-8' ?>
    <Response>
        <EnterCCS queueType='2' timeout='30' queueTimes='1'
agenthangupurl='agenthangup'>
            <Play>waitmusic.wav</Play>
        </EnterCCS>
        <Redirect tag='2'>transferAgentFail</Redirect>
    </Response>
```

如果用户按其他键的响应包体

```
<?xml version='1.0' encoding='UTF-8' ?>
    <Response>
        <Hangup/>
    </Response>
```

说明 主命令为进入队列命令 用户按 1 进入 1 队列 按 2 进入 2 队列 嵌套 waitmusic.wav 放音，如果有座席空闲则转到座席，座席接听并挂机后回调 agenthangup，座席超时未接听发话则跳转到 transferAgentFail。

f.IVR 座席未接听业务接口

```
public String transferAgentFail(HttpServletRequest req)
```

响应包体

```
<?xml version='1.0' encoding='UTF-8' ?>
    <Response>
        <Get action=' getchinesequeue ' numdigits='1' timeout='20'>
            <Play loop='-1'>menu.wav</Play>
        </Get>
        <Play>timeoutbye.wav</Play>
        <Hangup/>
    </Response>
```

说明：主命令为按键命令并且嵌套 menu.wav 的放音提示，提示用户是否继续排队，在

用户按键一个就回调通知到 dtmfreport 的地址，如果用户没有按键就会 20 秒超时后执行副命令，播放 timeoutbye.wav 的提示放音后挂机，通过判断 tag 的值区分是 1 队列还是 2 队列，跳转到不同的地址继续排队。

g.IVR 获取是否继续排队按键接口

```
public String getchinesequeue(HttpServletRequest req)
```

响应包体

```
<?xml version='1.0' encoding='UTF-8' ?>
    <Response>
        <EnterCCS queueType='1' timeout='30' queueTimes='1'
agenthangupurl='agenthangup'>
            <Play>waitmusic.wav</Play>
        </EnterCCS>
        <Hangup/>
    </Response>
```

说明：通过此命令让用户继续排队

h.IVR 座席挂机业务接口

```
public String agenthangup(HttpServletRequest req)
```

响应包体

```
<?xml version='1.0' encoding='UTF-8' ?>
    <Response>
        <Get action='pingjia' numdigits='1' timeout='20'>
            <Play loop='-1'>pingjia.wav</Play>
        </Get>
        <Play>bye.wav</Play>
        <Hangup/>
    </Response>
```

说明：获取用户按键，让用户对座席进行评价

i.IVR 评价业务接口

```
public String pingjia(HttpServletRequest req)
```

响应包体

```
<?xml version='1.0'
encoding='UTF-8' ?><Response><Play>thank.wav</Play><Hangup/></Response>
```

说明：得到用户的按键，放音提示完成后挂机

j.IVR 结束业务接口

```
public String stopservice(HttpServletRequest req)
```

响应包体

```
<Response>
    <CmdNone/>
</Response>
```

说明：主命令为空命令，因为结束业务了不需要返回其他的 IVR 命令。

K.IVR 坐席状态改变接口

```
public String agentstate(HttpServletRequest req)
```

响应包体

```
<Response>
    <CmdNone/>
</Response>
```

说明：坐席状态改变不需要后续命令，返回空命令即可，如果坐席接听后状态变化为 0 准备中需要通过 AgentReady 接口让坐席准备就绪。如果想实现坐席自动准备就绪，可以在坐席状态改变为 0 准备中的时候响应坐席准备就绪命令，坐席其他状态不需要后续业务直接返回空命令。

响应坐席准备就绪命令示例：

```
<Response>
<Appid>"+appid+"</Appid>
<AgentReady agentid='"+agentid+"'>
</Response>
```

3.Demo 的使用

- 1.调用 CreateQueue 接口，分别创建 1 队列和 2 队列；
- 2.调用 AgentOnWork 接口，分别在 1 队列和 2 队列上班座席，座席号码可以写控制台里配置的测试号码，并且座席状态设置为 1 准备就绪状态，如果设置坐席状态 0 准备中，可以通过 AgentReady 接口设置坐席准备就绪；
- 3.在座席接电话后座席状态会变成 0，需要调用 AgentReady.php 文件将座席重新变为准备就绪状态；
4. CCPRestSDK.php 为封装好的座席上班接口、座席准备就绪接口、创建队列接口和 IVR 外呼接口。

4. 参考文档

- (1) rest 座席管理接口文档

<http://docs.yuntongxun.com/index.php/%E5%BA%A7%E5%B8%AD%E7%AE%A1%E7%90%86>

- (2)IVR 基础命令接口文档：

http://docs.yuntongxun.com/index.php/IVR_%E5%9F%BA%E7%A1%80%E5%91%BD%E4%BB%A4

- (3)IVR 基础通知消息接口文档

http://docs.yuntongxun.com/index.php/IVR_%E5%9F%BA%E7%A1%80%E9%80%9A%E7%9F%A5%E6%B6%88%E6%81%AF

- (4)IVR 座席命令接口文档

http://docs.yuntongxun.com/index.php/IVR_%E5%BA%A7%E5%B8%AD

- (5) IVR 座席通知消息接口文档

http://docs.yuntongxun.com/index.php/IVR_%E5%BA%A7%E5%B8%AD%E9%80%9A%E7%9F%A5%E6%B6%88%E6%81%AF