**上上签API接口文档**

BestSign OpenAPI V3\_2.26.1

# 目录

目录 1

一、API简介 5

1、 注册用户 7

接口调用流程 7

2、快捷签署（自动签） 7

接口调用流程 7

3、手动签署 8

接口调用流程 8

4、 PDF签名图片尺寸、位置的计算方法（必读） 8

4.1、图片尺寸的计算方法 9

4.2、位置坐标的计算方法 9

二、API协议须知 10

1、 调用地址（host） 10

1.1、测试环境 10

1.2、正式环境（生产环境） 10

2、调用流程 11

3、请求类型 11

1）、POST/GET 11

2）、字符集 11

3）、JAVA的特殊处理 11

4、请求格式 11

5、生成签名（计算sign参数） 12

1）、先得到签名原文字符串 12

A）、POST方法原文字符串示例 13

B）、GET方法原文字符串示例 13

2）、签名计算方法（rsa算法） 13

6、接口响应 14

7、异步通知 15

8、返回消息体加密说明 18

使用方法 18

加密原理 18

三、标配API功能列表 20

1、用户服务 20

1.1、注册个人用户并申请证书 20

1.2、注册企业用户并申请证书 22

1.3、查询证书编号 27

1.4、查询个人用户证件信息 28

1.5、查询企业用户证件信息 29

1.11、异步申请状态查询 31

1.12、获取证书详细信息 32

2、签名/印章图片服务 34

2.1、生成用户签名/印章图片 34

2.2、上传用户签名/印章图片 35

2.3、下载用户签名/印章图片 36

3、文件存储服务 37

3.1、上传合同文件 37

3.3、为PDF文件添加元素 39

3.4、下载文件 40

3.5、PDF文件验签 41

4、单文档合同服务 43

4.1、上传并创建合同 43

4.2、签署合同（即自动签） 45

4.3、发送合同（即手动签，指定图片大小） 48

4.4、撤销合同 53

4.5、锁定并结束合同 54

4.6、查询合同信息 55

4.7、获取合同签署参数 57

4.8、查询合同签署者状态 59

4.9、下载合同文件 60

4.10、获取预览页URL 60

4.11、创建合同 62

4.21、模版签署功能 63

4.21.0、获取模版变量 67

4.21.1、通过模版生成合同文件 68

4.21.2、通过模版创建合同 70

4.21.3、用模版变量签署合同 72

4.21.4、获取模版信息 73

4.21.5、用模版变量的手动签 77

4.21.6、获取创建模版的地址 80

4.21.7、获取编辑模版的地址 81

4.21.8、获取开发者模版列表 82

4.21.9、预览模版 84

4.22、生成合同附页 85

4.23、下载合同附页文件 86

4.24、关键字定位签署合同 86

4.25、在线验签（通过合同ID和哈希值） 89

4.26、获取存证页URL 90

12、支付服务 91

12.1、微信支付 91

12.1、查询支付订单 92

五、选配API功能列表 94

1、用户服务 94

1.2、注册企业用户并申请证书（支持CFCA企业证书） 94

1.6、注册用户 99

1.7、设置个人用户证件信息 100

1.8、设置企业用户证件信息 101

1.9、同步申请数字证书 104

1.13、申请专用数字证书 105

1.14、吊销专用数字证书 106

3、文件存储服务 107

3.2、转换文件格式 107

4、单文档合同服务 108

4.2、签署合同（即自动签，每个位置可用不同的图片） 108

4.12、单个添加签署者 112

4.13、批量添加签署者 113

4.14、设置合同签署参数 114

4.15、结束合同 116

4.16、骑缝章签署 117

4.17、删除合同文件 118

4.18、获取合同手动签署页URL 119

4.19、拒绝签署 121

4.20、团签 122

4.27、跨开发者签署模式 125

4.27.1、获取跨开发者签署token 125

4.27.2、跨开发者签署 127

4.27.3、跨开发者下载 128

4.28、合同批量签署（即批量自动签） 129

4.29、合同关键字批量签署（即批量自动签） 130

4.30、WPS专用签署合同 132

4.31、一次性签署合同 136

4.32、上传合同文件到公有云 140

4.33、锁定并结束合同（自定义证书） 141

4.34、上传合同签署照片 142

4.35、ISV创建合同 143

4.36、带短信校验的自动签 145

4.37、合同PDF文件上添加附件 148

5、文件处理工具 150

5.1、PDF关键字查找位置 150

5.2、获取PDF文件预览图片 152

6、通知服务 153

6.1、发送签署合同验证码 153

6.2、校验签署合同验证码 154

6.3、生成短链接 155

6.4、发送通知短信 156

7、自定义合同编号的本地demo 157

一）、合同编号转换的通用方法 158

1）、建立合同编号映射 158

2）、查询合同编号 159

二）、本地demo方法列表 161

7.1、通过URL创建合同 161

7.2、签署合同 163

7.4、锁定并结束合同 165

7.5、下载合同文件 166

7.6、生成短链接 166

7.8、拒绝签署 167

7.9、单个添加签署者 168

7.10、批量添加签署者 169

7.11、设置合同签署参数 170

7.12、获取手动签署URL 172

8、多文档合同服务 174

8.1、创建合同目录 174

8.2、向目录添加合同文件（根据fid或者fdata添加） 176

8.3、获取合同列表 178

8.4、获取多文档手动签署URL 180

8.5、获取预览合同URL 185

8.6、签署合同（即自动签） 186

8.7、锁定并结束目录 188

8.8、结束目录 190

8.9、生成目录的附页 191

8.10、下载目录附页文件 192

8.11、向目录添加合同文件（根据contractId添加） 192

8.12、使用模版变量的多文档手动签 194

10、不允许WEB用户查看、下载接口合同 198

4.1、上传并创建合同 198

4.11、创建合同 200

4.21.1、通过模版生成合同文件 201

8.2、向目录添加合同文件 203

8.11、向目录添加合同文件（根据contractId添加） 205

11、 国密证书服务 207

11.1、申请国密数字证书 207

11.2、查询国密证书编号 208

11.3、用国密证书签署合同（即自动签） 209

11.4、用模版变量和国密证书签署合同 212

11.5、注册个人用户并申请国密证书 214

11.6、注册企业用户并申请国密证书 216

11.8、用国密证书的手动签 221

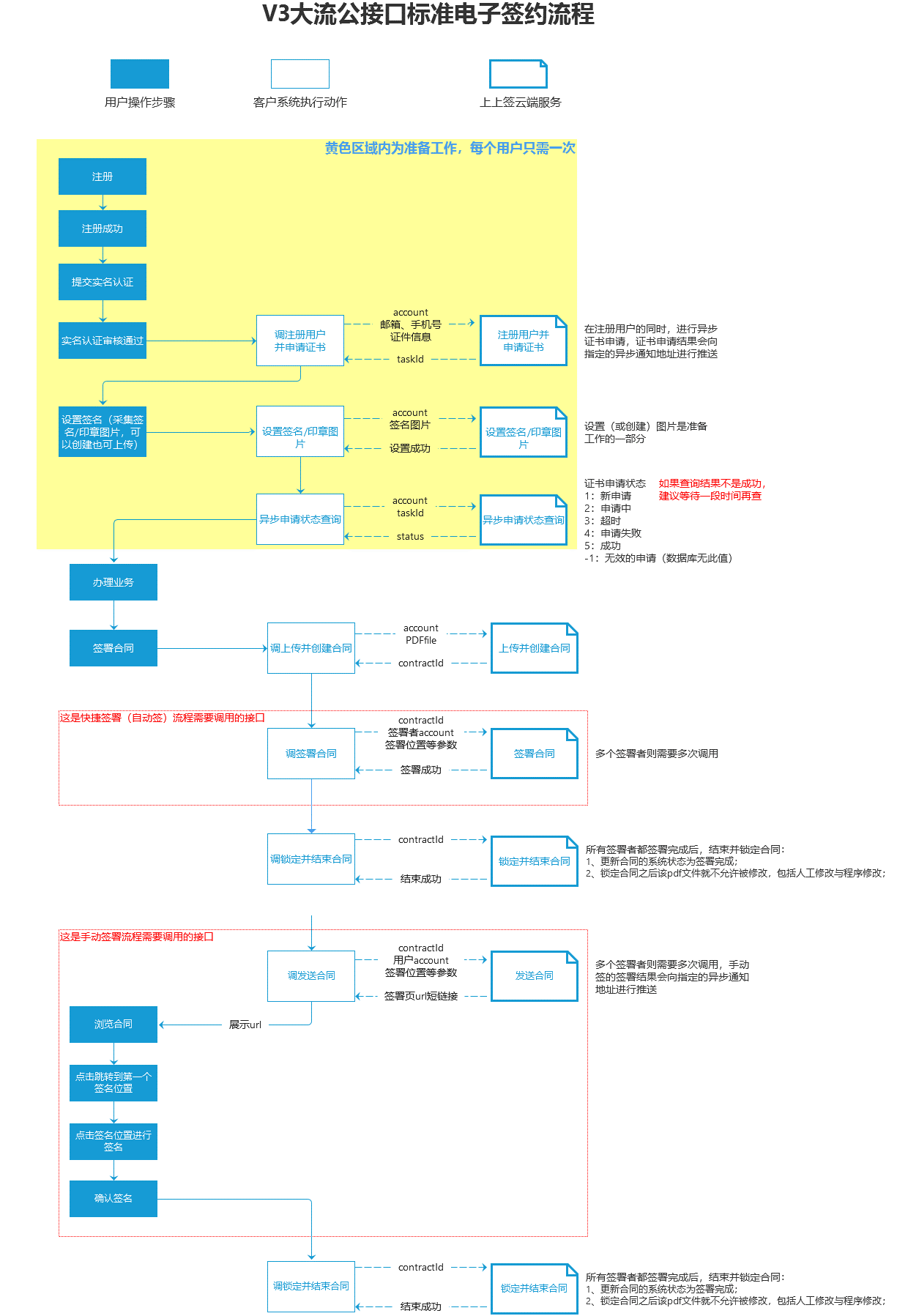
11.9、用国密证书的多文档手动签 226

六、附录：返回信息编号列表 230

文档修订记录 233

# 一、API简介

根据不同的用户接入要求，上上签OpenAPI提供两种签署模式：快捷签署（又叫自动签，由系统自动签署，用户无感知）、手动签署（体现签署操作的仪式感）。“快捷签署”与“手动签署”可以混搭，比如同一份合同中，甲用“快捷签署”，乙用“手动签署”。总体流程如下图所示：



## 注册用户

使用上上签OpenAPI进行合同的发送、签署等服务的用户，必须要先注册一个上上签的用户，此后的所有操作都使用该用户account（用户唯一标识）进行。

### 接口调用流程

相关接口如下：

|  |  |
| --- | --- |
| [注册个人用户并申请证书/user/reg](#_1.1、注册用户) | 包含：“注册用户、设置证件信息、异步申请证书”3个功能，证书是否申请成功需要通过“异步申请状态查询”来确认 |
| [注册企业用户并申请证书/user/reg](#_1.2、注册企业用户并申请证书) | 包含：“注册用户、设置证件信息、异步申请证书”3个功能，证书是否申请成功需要通过“异步申请状态查询”来确认 |
| [创建用户签名或印章图片/signatureImage/user/create](#_2.1、生成用户签名/印章图片) | 二选一，要么创建要么上传，创建与上传会相互覆盖默认图片 |
| [上传用户签名或印章图片/signatureImage/user/upload](#_2.2、上传用户签名/印章图片_1) | 二选一，要么创建要么上传，创建与上传会相互覆盖默认图片 |
| [异步申请状态查询/user/async/applyCert/status/](#_1.11、异步申请状态查询) | 在申请证书之后、签署之前请先使用此接口确认证书是否申请成功，避免在签署时出错 |

## 2、快捷签署（自动签）

用户无感知的快捷签署，又称为“自动签”、“静默签”等。

### 接口调用流程

相关接口如下：

|  |  |
| --- | --- |
| [上传并创建合同/storage/contract/upload](#_4.1、上传并创建合同) | 上传PDF文件并创建合同 |
| [签署合同/storage/contract/sign/cert](#_5.5、设置合同签署参数) | 自动签署 |
| [锁定并结束合同/storage/contract/lock/](#_4.4、锁定并结束合同) | 当合同的所有签署人都签署完毕后，调用此接口结束合同的整个签署过程，结束后将不允许进行任何签署操作 |

## 3、手动签署

用户有感知的签署，有手写签名、校验短信验证码等交互动作，体现签署的仪式感。

### 接口调用流程

相关接口如下：

|  |  |
| --- | --- |
| [上传并创建合同/storage/contract/upload](#_5.1、创建合同) | 上传PDF文件并创建合同 |
| [发送合同/contract/send](#_4.3、发送合同) | 为某个手动签用户生成签署链接，可将此链接发送给用户，打开此链接进入手动签页面，在页面上查看合同并完成签署，体现签署的仪式感 |
| [锁定并结束合同/storage/contract/lock](#_4.4、锁定并结束合同) | 当合同的所有签署人都签署完毕后，调用此接口结束合同的整个签署过程，结束后将不允许进行任何签署操作 |

## PDF签名图片尺寸、位置的计算方法（必读）

尺寸单位换算（来源百度）：1厘米≈28.35磅

常用规格印章尺寸：

3.5厘米≈99磅

4厘米≈113磅

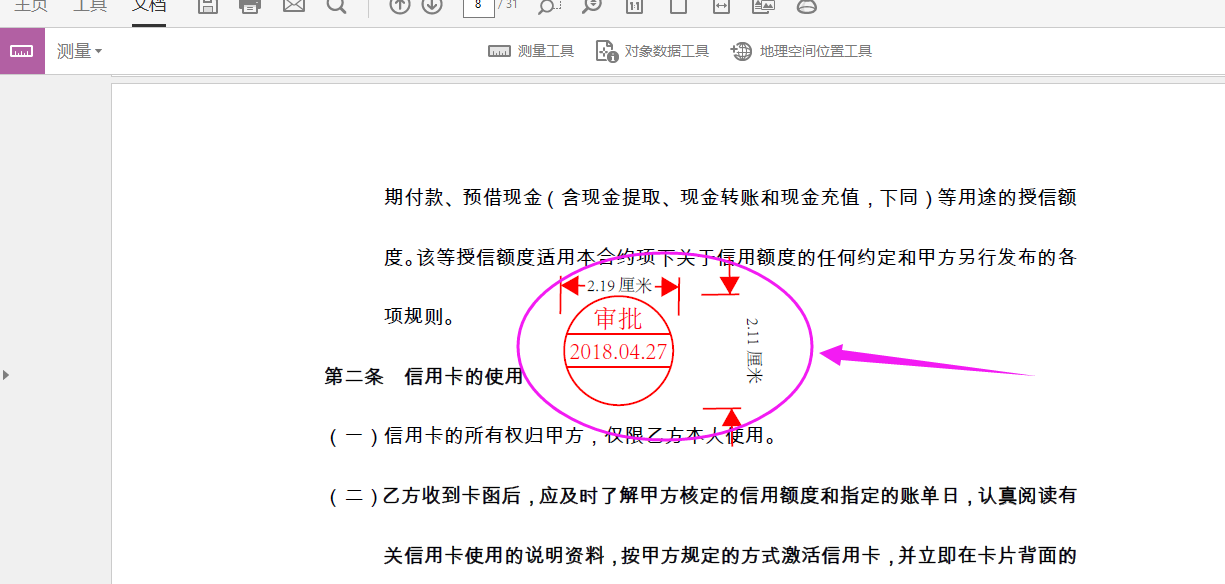
4.2厘米≈119磅

5厘米≈142磅

5.8厘米≈164磅

A4纸宽20.9厘米≈593磅，A4纸高29.7厘米≈842磅

### 4.1、图片尺寸的计算方法



如上图所示，一个图片在一份PDF想要显示的大小是有一个具体尺寸的，比如常见的圆形公章有3.5厘米×3.5厘米、4.0厘米×4.0厘米等规格的尺寸。而计算机在处理图片时，需要用到的图片尺寸单位为磅值，换算比例（来源百度）为1厘米≈28.35磅。

如果想要一个3.5厘米×3.5厘米公章图片在PDF上体现为真实大小，那么在接口参数中就应该这样传（取四舍五入后的整数）：3.5厘米×28.35磅/厘米=99.225磅≈99磅：

**"signatureImageWidth"**: "99"**,**

**"signatureImageHeight"**: "99"

### 4.2、位置坐标的计算方法

在上上签的坐标体系中，以页面的左上角为坐标原点、以签名/印章图片的左上角来计算图片所在的位置坐标，位置坐标为一个带小数点的比例数值，因此按像素单位计算、按磅值单位计算、或者按厘米单位计算都可以，以下用厘米单位举例：



如上图，印章图片左上角距离左侧页边距为8.37厘米、距离上侧页边距3.95厘米，页面为A4纸大小，宽20.9厘米，高29.7厘米，那么这个图片在页面上坐标的计算方法为：

X=8.37厘米÷20.9厘米=0.400478

Y=3.95厘米÷29.7厘米=0.132997

在参数中应该这样传坐标：

**"x"**: "0.400478"**,**

**"y"**: "0.132997"**,**

# 二、API协议须知

上上签OpenAPI基于HTTPS协议，您可以自行封装请求进行调用，以下对自行封装请求进行说明。

## 调用地址（host）

### 1.1、测试环境

**server-host**：用于调用上上签云端接口，是上上签配置的host地址，开发者不可变更。

https://apistore.huaweicloud/openapi/v2

请注意：在将host与接口方法拼接到一起时，v2后面的斜杠“/”只需要1个，如果有2个“/”会出现404错误请求失败。

### 1.2、正式环境（生产环境）

**server-host**：用于调用上上签云端接口，是上上签配置的host地址，开发者不可变更。

https://apistore.huaweicloud/openapi/v2

请注意：在将host与接口方法拼接到一起时，v2后面的斜杠“/”只需要1个，如果有2个“/”会出现404错误请求失败。

## 2、调用流程

## 3、请求类型

### 1）、POST/GET

POST方式提交的请求，BODY一律使用JSON格式。请求JSON格式时，Headers里需要设置Content-Type为application/json。

GET方式提交的请求，Headers和Request Body都不需要。

### 2）、字符集

字符编码设置为UTF-8。

### 3）、JAVA的特殊处理

HttpClientUtil.httpPostRequest 这个方法里面改一下  
ByteArrayEntity byteArrayEntity = new ByteArrayEntity(JSON.toJSONString(params).getBytes("UTF-8"));  
httpPost.setEntity(byteArrayEntity);  
把StringEntity那里，改成ByteArrayEntity

系统底层好像会把string转换成charAt，所以utf8的中文就不知道它会怎么处理，开发人员在外面把它转成utf8的字节数组，不让系统自动转。

## 4、请求格式

**GET/POST** [host]/**Service**/**Method**/?developerId=开发者编号&rtick=unix时间戳+4位随机数&signType=签名串计算方法&sign=签名串&[其他URL参数......]

如果是POST请求，把请求参数，以json格式放body里，同时设置Content-Type为application/json。

一般的请求URL格式如下：

https://openapi.bestsign.info/openapi/v2/user/reg/?**developerId**=3333&**rtick**=15066659931690&**signType**=rsa&**sign**=OSIBr6v2u121r8QwE7mBRhC9iuRUWw24W8o1%2Fm87zqs6s1fDEVwAjz8S7939p2eXAWpcnuOsPU13eD1PklLL7EiyQBIXAaIQBUXrhjqX%2BNpG3qdvuSkbY5g%2BRdlLHXS3sR%2FeXBtIvCJAJY6yM9iA%2Fm9gtY3mJyrGh83APb3m5Lw%3D

其中，**developerId, rtick, sign, signType**这几个参数是每一个接口都必须有的参数。url里的其他参数，跟具体的接口相关。

## 5、生成签名（计算sign参数）

### 1）、先得到签名原文字符串

（1）查看您申请的开发者编号（developerId）。假设您的developerId为aaaaaa，rtick为14840209310001。

（2）将url中，**需要参与签名计算的各url参数**，按参数名的**字典序**进行排序，并将**排序后**的各参数按“**参数名1=参数值1参数名2=参数值2......参数名n=参数值n**”的格式连接成一个字符串（各参数之间，不要用&分隔） 。

（3）将请求的HTTP Request Path，如/openapi/v2/user/reg，附加到这个字符串后面。

（4）如果是POST请求，有Request Body，将Request Body的md5值，再附加到这个字符串后面得到最终的签名原文字符串。

（5）对签名原文字符串进行签名计算，签名计算方法为rsa算法，通过url参数signType指定。

#### A）、POST方法原文字符串示例

假设调用注册用户接口，POST请求/openapi/v2/user/reg，使用rsa算法计算签名，时间戳为14840209310001，Request Body为：

**{**"mail"**:**"test@bestsign.cn"**,**"type"**:**"2"**}**

则请求的URL为：

**/openapi/v2/**user**/**reg**/?**developerId**=**aaaaaa**&**rtick**=**14840209310001**&**signType=rsa

按如下步骤得到签名原文字符串：

1. 先把需要签名计算的参数进行字典序排序，得到一个字符串：

developerId=aaaaaartick=14840209310001signType=rsa

b. 把请求path附加到字符串后面：

developerId=aaaaaartick=14840209310001signType=rsa/openapi/v2/user/reg/

c. 计算request body的md5值(假设我们的request body的md5值是fb5939fd51b1a929a6e2b75b187827f7)，附加到字符串后面，得到最终的签名原文字符串：

developerId=aaaaaartick=14840209310001signType=rsa/openapi/v2/user/reg/fb5939fd51b1a929a6e2b75b187827f7

#### B）、GET方法原文字符串示例

因为GET方法没有Headers和Request Body，因此原文字符串与POST方法略有差别。以/openapi/v2/signatureImage/user/download为例：

参数按照参数名进行字典排序后得到签名：

1. 先把需要签名计算的参数进行字典序排序，得到一个字符串：

account=136developerId=1709264203795136512imageName=rtick=14928387557256980signType=rsa

b. 把请求path附加到字符串后面，得到签名原文字符串：

account=136developerId=1709264203795136512imageName=rtick=14928387557256980signType=rsa/openapi/v2/signatureImage/user/download/

### 2）、签名计算方法（rsa算法）

使用rsa算法对这个签名原文字符串进行签名计算。

URL参数中的signType需要设为 rsa，然后使用私钥对这个签名原文字符串进行rsa加密。

用伪码表示这个过程：

rsa("developerId=aaaaaartick=14840209310001signType=rsa/openapi/v2/user/reg/fb5939fd51b1a929a6e2b75b187827f7")

得到sign： ECaLEwc1Wq+XzaGDO5I51uL35Ad+LRpUcunbVj5WGQzCAH7SBIH6EBoZaYLwo5Kicbv/Iy1t8PveILhlUDDJQL+AgB6sUfSdrT6j6Xk7U9Na/VQAsz3/DMmE8hxUK0jDlO196egh5WsLZdy5Lg9XJqVclCou4VTq4xWYvaMVTIw=

注：

①使用的rsa算法为SHA1withRSA；

②rsa之后需要进行一次base64Encode，之后再使用UTF-8进行一次urlEncode（因为base64Encode之后的字符串可能会带有“/”等特殊符号），得到的字符串才能用于http请求的url参数。

③特别注意：如果拼装到http请求url里的参数带有“/”等特殊符号的，则该参数需要先进行一次urlEncode，得到的字符串才能拼到http请求url里。对参数的urlEncode只需做一遍即可，重复urlEncode之后使用会出错。

## 6、接口响应

业务正常返回HTTP 200。

API的响应包的Body部分采用JSON格式编码，因此需要对返回包进行JSON解析。

如果是下载类的请求（即响应的Body是数据流，不是JSON格式），无法使用JSON输出响应。业务正常时会正常返回数据流，当发生业务异常时，服务端会返回5xx类型的HTTP状态码，同时会将errno和errmsg放到Header中输出。

其他类型的请求，将会以JSON格式输出数据。输出格式：

**{**

**"errno"**: 0**,** // 错误码, 如果为0表示成功没有错误

**"data"**: **{ },** // 输出数据 （如果有的话，具体内容和每一个具体的接口相关）

**"cost": 233,** // 接口处理花费的时间

**"errmsg"**: "" // 错误描述，结合errno对照附录

**}**

其中：

* errno返回0则为正常，errno返回错误码则为异常，此时观察具体的errmsg具体分析。errno与errmsg请参阅本文《四、附录：返回信息编号列表》，返回信息编号列表持续更新。
* cost是本接口的响应时间，表示上上签从收到请求到给出响应所经历的时间（上上签内部执行的时间），单位是毫秒，数字格式。
* data是接口的具体返回参数，参见具体接口。

## 7、异步通知

有的开发者需要上上签主动将结果通知给开发者，因此我们需要开发者向上上签提供一个pushUrl，开发者将这个pushUrl告知上上签，由上上签设置到开发者配置中，并且将推送消息格式及accessKey告知开发者。

此pushUrl的基本要求：

（1）必须支持http或者https格式；

（2）请求类型必须为POST；

（2）能够接收消息并返回http状态为200；

（3）必须是一个完整的URL（如：https://www.yourdomain.com/message/receiveBestsign.do），不能是一个相对路径（/message/receiveBestsign.do）；

（4）安全要求：使用上上签实施提供的accessKey加解密。

1、异步通知针对下列接口有效：

"1.1、注册个人用户并申请证书"——异步申请证书的申请结果会异步推送；

"1.2、注册企业用户并申请证书"——异步申请证书的申请结果会异步推送；

"4.3、发送合同"——将签署链接给用户进行手动签署的签署成功结果会异步推送；

"4.21.6、获取创建模版的地址"——创建模版保存后的结果会异步推送；

"4.21.7、获取编辑模版的地址"——编辑模版保存后的结果会异步推送；

"12.1、微信支付"——通过微信扫码支付后的支付结果会异步推送；

2、推送消息格式（httpbody）：

//签署合同的推送消息格式如下：

**{**

**"action"**: "signContract"**,** //针对的事件，signContract是签署合同

**"params"**: **{**

**"contractId"**: "1093274328532857"**,** //合同编号

**"account"**: "mlimd@163.com"**,** //签署者用户标识

**"signerStatus"**: 2**,** //签署状态：（1：正在进行中；2：已完成；3：已拒绝并取消；）

**"errMsg"**: "success"**,** //签署成功为success，签署失败为相应的错误信息

**"sid"**: "001" //业务流水号，如果在send接口中提交了则会返回，没提交则不返回

**}**

**}**

//异步申请证书的推送消息格式如下

**{**

**"action"**: "certApply"**,** //针对的事件，certApply是申请证书

**"params"**: **{**

**"certType"**: "ZJCA"**,**

**"cert"**: "ZJCA-31-20171201174841514-03517"**,** //证书编号

**"message"**: ""**,**

**"account"**: "ent1512121718841"**,** //申请证书的用户标识

**"taskId"**: "151212172101000002"**,** //异步申请队列的序号

**"status"**: "5"

**}**

**}**

//创建模版的推送消息格式如下

**{**

**"action"**: "createTemplate"**,** //针对的事件，createTemplate是创建模版

**"params"**: **{**

**"tid"**: "2489738745798734"**,** //开发者模版编号

**"account"**: "13897423874"**,** //证书编号

**"sid"**: "47283u78347846" //业务流水号

**}**

**}**

//编辑模版的推送消息格式如下

**{**

**"action"**: "modifyTemplate"**,** //针对的事件，modifyTemplate是创建模版

**"params"**: **{**

**"tid"**: "2489738745798734"**,** //开发者模版编号

**"account"**: "13897423874"**,** //证书编号

**"sid"**: "47283u78347846" //业务流水号

**}**

**}**

//支付接口的推送消息格式如下

**{**

**"action"**: "onlinePay"**,**

**"params"**: **{**

**"orderId"**: "2130985035547807747"**,**

**"paymentMode"**: "WECHATPAY"**,**

**"contractId"**: "2371924"**,**

**"completeDateTime"**: "2018-11-22 14:28:19 "**,**

**"orderAmount"**: "0.4"**,**

**"payStatus"**: "SUCCESS"

**}**

**}**

3、推送消息的加密机制：

推送消息的headers中会含有本次推送的加密信息，用于校验。

**"headers"**: **{**

**"rtick"**: "15004551687129438"**,**

**"sign"**: "d517a52a9910f164ec42bf8c1a7803b2"

**}**

**rtick ----**时间戳，unix时间戳**+**4位随机数

**sign ----**就是sign，sign**=**md5**(**md5**(**httpbody**)+**rtick**+**accessKey**)**

## 8、返回消息体加密说明

有些开发者对数据的保密性有更高的要求，因此上上签在返回消息时也进行加密加签。

### 使用方法

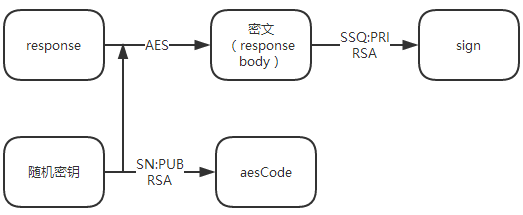
开发者若需要返回的消息加密，需要在request的Headers添加属性“encryptResponse”，仅在“encryptResponse”值为1时表示需要加密返回。目前支持加密的response body数据类型是byte[] 、String 、JSONObject这三种。

### 加密原理

在此加密加签过程中，需要用到：

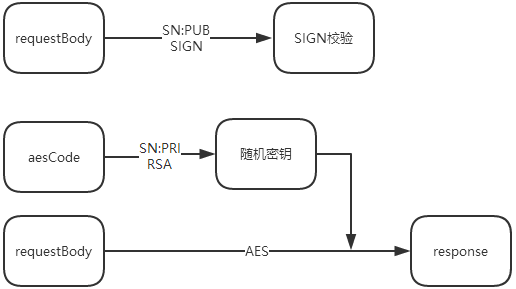
* 上上签的私钥（SSQ:PRI）——上上签生成，由上上签保管
* 上上签的公钥（SSQ:PUB）——上上签生成，提供给开发者，做验签用
* 随机密钥——即accessKey，上上签生成（可登录开放平台获取），用作AES加密，并提供给开发者用作AES解密
* 开发者的公钥（SN:PUB）——开发者生成，提供给上上签，做验签用
* 开发者的私钥（SN:PRI）——开发者生成，由开发者保管

1、加密原理



使用随机密钥对接口的response进行AES加密，然后将密文放在response body中，并将body使用上上签私钥进行RSA加签得到sign用于验签，同时将随机密钥使用开发者的公钥RSA加密得到aesCode，aesCode与sign都放在response header中。

2、解密步骤



开发者接收到上上签的response之后，使用上上签公钥对header中的sign进行验签，验签通过，则使用开发者私钥对aesCode进行解密得到随机密钥，再用随机密钥对body中的密文进行解密得到真正的接口response。

3、返回加密报文示例

HTTP**/**1.1 200 OK

Server**:** Apache**-**Coyote**/**1.1

aesCode**:** fT27wCCS6GnDlAgiCeHiUPTDR0XSJ**/**uINdNMq**+**DdLFkz4YORObEK7wrsu8xNf0ECYM**+**WZ3zhnEMd**/**IDQj0gcot7E**+**0b5Kjo7mIdkMNB9EFFFlOsBEbfjW1yFt2zJn738brkNDE25KdjeAvMnDS2O51tDm0XJPhDlH8QqaA8aSvg**=**

sign**:** ZMLGaDhvNyerdCXHm**/**28SvvGF3N46VTchSCAFISa5Y**+**cZUFDLKQkWay5Y**/**GhIAMs9Dh**/**z98NB5HMFdLPFTRNim46MKVZUNsEAHuZNoowSyX82R**/**SmSq0RkEn4bTQRru3nniKyMBkKU**/**pXvGpGzn0a**/**g0KmRRrMrQ**+**2y3xdfnybU**=**

Expires**:** **-**1

Pragma**:** no**-**cache

CacheControl**:** no**-**cache

Content**-**Length**:** 88

Date**:** Thu**,** 02 Nov 2017 02**:**48**:**33 GMT

Content**-**Type**:** application**/**json**;**charset**=**UTF**-**8

RESPONSE BODY**:**

FmCKgiFAuiasYrRiQolCtR3NbZZJGJ020esImdmg**/**TyzcGL7ftqR8GoOkRSSNUYMq6FH6zlOVTb8C0uPqe1f0Q**==**

# 三、标配API功能列表

## 1、用户服务

### 1.1、注册个人用户并申请证书

说明：创建一个上上签个人用户，并提交用户的证件信息，为该用户申请数字证书。后续有关该用户的所有操作都使用该用户的account。

由于证书是异步申请，不会立即返回证书申请结果，所以在签署之前，需要确认一次该用户的证书是否申请成功，调用接口“1.11、异步申请状态查询”确认。

Request Path：[**server-host**]/user/reg/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | | | |
| **account** | 用户唯一标识 | Y |  | 用户的唯一标识，可以是邮箱、手机号、证件号等不限 | | | | |
| **name** | 用户名称 | Y | 100 | 必须和证件上登记的姓名一致 | | | | |
| **userType** | 用户类型 | Y | 1 | 1表示个人 | | | | |
| **mail** | 用户邮箱 | N | 50 |  | | | | |
| **mobile** | 用户手机号 | N | 20 |  | | | | |
| **credential** | 用户证件信息对象 | Y |  | 用户证件信息对象，json格式么具体参数如下： | | | | |
| **参数** | **说明** | **是否必填** | **最大长度** |  |
| **identity** | 用户证件号 | Y | 30 | 必须和证件上登记的号码一致 |
| **identityType** | 用户证件类型 | N | 2 | 默认为“0”, 0-居民身份证  1-护照  B-港澳居民往来内地通行证  C-台湾居民来往大陆通行证  E-户口簿  F-临时居民身份证 |
| **contactMobile** | 联系手机 | N | 20 |  |
| **contactMail** | 联系邮箱 | N | 50 |  |
| **province** | 省份 | N | 32 |  |
| **city** | 城市 | N | 50 |  |
| **address** | 地址 | N | 100 |  |
| **applyCert** | 是否申请证书 | Y | 1 | 申请填写为1 | | | | |

**示例：**

**{**

**"account"**: "test@bestsign.cn"**,**

**"mail"**: "test@bestsign.cn"**,**

**"mobile"**: "13800001234"**,**

**"name"**: "张三"**,**

**"userType"**: "2"**,**

**"credential"**: **{**

**"identity"**: "100123199001010011"**,**

**"identityType"**: "0"**,**

**"contactMail"**: "test@bestsign.cn"**,**

**"contactMobile"**: "13800001234"**,**

**"province"**: "浙江省"**,**

**"city"**: "杭州市"**,**

**"address"**: "xx路xx号"

**},**

**"applyCert"**: "1"

**}**

**响应（RESPONSE）：**

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **taskId** | 异步申请证书队列中的任务编号，在24小时内可用于查询异步申请状态，**taskId**过24小时后就失效 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"taskId"**: "170217170217942"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 1.2、注册企业用户并申请证书

说明：创建一个上上签企业用户，并提交用户的证件信息，为该用户申请数字证书。后续有关该用户的所有操作都使用该用户的account。

由于证书是异步申请，不会立即返回证书申请结果，所以在签署之前，需要确认一次该用户的证书是否申请成功，调用接口“1.11、异步申请状态查询”确认。

Request Path：[**server-host**]/user/reg/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | | | |
| **account** | 用户唯一标识 | Y |  | 用户的唯一标识，可以是邮箱、手机号、证件号等不限 | | | | |
| **name** | 企业名称 | Y | 100 | 必须和企业证件上登记的名称一致  如个体工商户在营业执照上无企业名称的，请填“经营者”名字 | | | | |
| **userType** | 用户类型 | Y | 1 | 2表示企业 | | | | |
| **mail** | 用户邮箱 | N | 50 |  | | | | |
| **mobile** | 用户手机号 | N | 20 |  | | | | |
| **credential** | 企业证件信息对象 | Y |  | 企业证件信息对象，json格式  三证合一的统一社会信用代码这样传：regcode = "统一社会信用代码"orgcode = "统一社会信用代码"taxcode = "统一社会信用代码"老三证这样传：regcode = "工商注册号"orgcode = "组织机构代码证"taxcode = "税务登记证"个体户这样传：regcode = "工商注册号"orgcode = ""taxcode = ""  具体参数如下： | | | | |
| **参数** | **说明** | **是否必填** | **最大长度** |  |
| **regCode** | 工商注册号 | Y | 30 | 企业用户设置 |
| **orgCode** | 组织机构代码 | Y | 30 | 企业用户设置  regCode和orgCode必须输入一个，否则异常返回：240006: regCode/orgCode must be have one |
| **taxCode** | 税务登记证号 | Y | 30 | 企业用户设置 |
| **legalPerson** | 法定代表人姓名 | Y | 50 | 法定代表人姓名或经办人姓名 |
| **legalPersonIdentity** | 法定代表人证件号 | Y | 30 | 法定代表人证件号或经办人证件号 |
| **legalPersonIdentityType** | 法定代表人证件类型 | Y | 2 | 法定代表人证件类型或经办人证件类型，与“**legalPersonIdentity**”要匹配，默认为“0”。  0-居民身份证  1-护照  B-港澳居民往来内地通行证  C-台湾居民来往大陆通行证  E-户口簿  F-临时居民身份证 |
| **legalPersonMobile** | 法定代表人手机号 | N | 20 | 法定代表人手机号或经办人手机号 |
| **contactMobile** | 联系手机 | Y | 20 | 联系手机必填，为CA年检抽查时联系使用 |
| **contactMail** | 联系邮箱 | N | 50 | **credential对象属性** |
| **province** | 省份 | N | 32 | **credential对象属性** |
| **city** | 城市 | N | 50 | **credential对象属性** |
| **address** | 地址 | N | 100 | **credential对象属性** |
| **applyCert** | 是否申请证书 | Y | 1 | 申请填写为1 | | | | |

**示例：**

**{**

**"account"**: "test@bestsign.cn"**,**

**"mail"**: "test@bestsign.cn"**,**

**"mobile"**: "13800001234"**,**

**"name"**: "浙江xx公司"**,**

**"userType"**: "2"**,**

**"credential"**: **{**

**"regCode"**: "5566789"**,**

**"taxCode"**: "5566789"**,**

**"orgCode"**: "5566789"**,**

**"contactMail"**: "test@bestsign.cn"**,**

**"contactMobile"**: "13800001234"**,**

**"legalPerson"**: "张三"**,**

**"legalPersonIdentity"**: "100123199001010011"**,**

**"legalPersonIdentityType"**: "0"**,**

**"legalPersonMobile"**: "17712341234"**,**

**"province"**: "浙江省"**,**

**"city"**: "杭州市"**,**

**"address"**: "xx路xx号"

**},**

**"applyCert"**: "1"

**}**

**响应（RESPONSE）：**

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **taskId** | 异步申请证书队列中的任务编号，在24小时内可用于查询异步申请状态，**taskId**过24小时后就失效 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"taskId"**: "170217170217942"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 1.3、查询证书编号

说明：查询某一用户的数字证书编号。

Request Path：[**server-host**]/user/getCert/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 查询哪个用户的证书就填该用户的唯一标识 |

**示例：**

**{**

**"account"**: "1674224491900698632"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **account** | 用户唯一标识 |
| **certId** | 证书编号 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"account"**: "1674224491900698632"**,**

**"certId"**: "CFCA-33-20171010142326279-12345"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 1.4、查询个人用户证件信息

说明：获取个人用户的证件信息。

Request Path：[**server-host**]/user/getPersonalCredential/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 查询哪个用户的证件信息就填该用户的唯一标识 |

**示例：**

**{**

**"account"**: "mlimd@163.com"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **account** | 用户唯一标识 |
| **identity** | 证件号 |
| **identityType** | 证件类型。默认为“0”, "0"表示身份证 |
| **name** | 姓名 |
| **contactMail** | 联系邮箱 |
| **contactMobile** | 联系手机 |
| **province** | 身份 |
| **city** | 城市 |
| **address** | 地址 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"account"**: "1238912387313"**,**

**"identity"**: "100123199001010011"**,**

**"identityType"**: "0"**,**

**"contactMail"**: "test@bestsign.cn"**,**

**"contactMobile"**: "13800001234"**,**

**"name"**: "张三"**,**

**"province"**: "浙江省"**,**

**"city"**: "杭州市"**,**

**"address"**: "xx路xx号"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 1.5、查询企业用户证件信息

说明：获取企业用户的证件信息。

Request Path：[**server-host**]/user/getEnterpriseCredential/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 查询哪个用户的证件信息就填该用户的唯一标识 |

**示例：**

**{**

**"account"**: "mlimd@163.com"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **account** | 用户唯一标识 |
| **regCode** | 工商注册号 |
| **taxCode** | 税务登记号 |
| **orgCode** | 组织机构代码 |
| **name** | 姓名/名称 |
| **legalPerson** | 法定代表人 |
| **legalPersonIdentity** | 法定代表人证件号 |
| **legalPersonIdentityType** | 法定代表人证件类型。默认为“0”, "0"表示身份证 |
| **legalPersonMobile** | 法定代表人手机号 |
| **contactMail** | 联系邮箱 |
| **contactMobile** | 联系手机 |
| **province** | 身份 |
| **city** | 城市 |
| **address** | 地址 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"regCode"**: "556789"**,**

**"taxCode"**: "556789"**,**

**"orgCode"**: "556789"**,**

**"legalPerson"**: "张三"**,**

**"legalPersonIdentity"**: "100123199001010011"**,**

**"legalPersonIdentityType"**: "0"**,**

**"legalPersonMobile"**: "17712341234"**,**

**"contactMail"**: "test@bestsign.cn"**,**

**"contactMobile"**: "13800001234"**,**

**"name"**: "浙江xx公司"**,**

**"province"**: "浙江省"**,**

**"city"**: "杭州市"**,**

**"address"**: "xx路xx号"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 1.11、异步申请状态查询

说明：

异步申请证书请求发送成功后，根据返回的taskId调用本接口查询申请结果。

Request Path：[**server**-host]/user/async/applyCert/status/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 需要查询证书申请状态的用户唯一标识 |
| **taskId** | 任务单号 | Y |  | 由异步申请数字证书接口返回得到，**taskId**在24小时内可用于查询 |

**示例：**

**{**

**"account"**: "mlimd@163.com"**,**

**"taskId"**: "170217170217942"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **status** | 证书申请状态  1：新申请  2：申请中  3：超时  4：申请失败  5：成功  -1：无效的申请（数据库无此值）  0：taskId不存在或已过期 |
| **message** | 状态描述 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"message"**: ""**,**

**"status"**: "5"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 1.12、获取证书详细信息

说明：根据某一用户的证书编号，获取证书的详细信息，此详细信息与查看pdf签名证书详情时看到的详细信息一致。

Request Path：[**server**-host]/user/cert/info/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 查询哪个用户的证书就填该用户的唯一标识 |
| **certId** | 证书编号 | Y | 10 | 通过“1.3、查询证书编号”接口查询到的certId，如CFCA-100123199001010011 |

**示例：**

**{**

**"account"**: "1674224491900698632"**,**

**"certId"**: "CFCA-100123199001010011"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **certId** | 证书编号，在上上签系统使用 |
| **serialNumber** | CA证书序列号，与查看pdf签名证书详情时看到的序列号一致 |
| **subjectDN** | 证书主题，即该证书代表的个人或企业的名称等信息 |
| **issuerDN** | 颁发机构 |
| **startTime** | 有效期开始时间 |
| **stopTime** | 有效期截止时间 |
| **revokedTime** | 吊销日期，正常使用中的证书无此项内容 |
| **revokedReason** | 吊销原因，正常使用中的证书无此项内容 |
| **status** | 状态码 1， -2 |
| **statusMsg** | 状态描述，1：激活， -2：吊销 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost"**: 88**,**

**"data"**: **{**

**"serialNumber"**: "68811245717"**,**

**"statusMsg"**: "激活"**,**

**"issuerDN"**: ""**,**

**"revokedTime"**: null**,**

**"startTime"**: "2017-08-09 14:27:51"**,**

**"stopTime"**: "2019-08-09 14:27:51"**,**

**"certId"**: "CFCA-33-20170809142751732-81453"**,**

**"revokedReason"**: null**,**

**"subjectDN"**: ""**,**

**"status"**: 1

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 241310**,**

**"cost"**: 88**,**

**"data"**: null**,**

**"errmsg"**: "using cert is not same with query cert"

**}**

## 2、签名/印章图片服务

### 2.1、生成用户签名/印章图片

说明：调用此接口可根据用户注册的名称、也可以根据指定的文本，系统生成用户签名图片或者印章图片（个人用户生成签名图片，企业用户生成印章图片），并且设置为用户默认签名。

Request Path：**[server-host]**/signatureImage/user/create/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **account** | 用户唯一标识 | Y | 创建哪个用户的签名/印章图片就填该用户的唯一标识。企业印章横向文默认为“电子签约专用章”。 |
| **text** | 签名/印章图片上生成的文本 | N | 可不填，系统默认使用证件信息上的名称 |
| **fontName** | 字体名称（仅针对个人类型账号有效） | N | 目前枚举值如下：  SimHei 黑体  SimSun 宋体  SimKai 楷体 |
| **fontSize** | 字号（仅针对个人类型账号有效） | N | 12~120，默认30，此参数影响签名字体的清晰度和签名图片大小，字号越高，字体显示越大，清晰度越高。  注：过小的字号在手动签的预览页面上显示会与实际大小有差别，但签署之后的PDF上的大小正常。 |
| **fontColor** | 字体颜色（仅针对个人类型账号有效） | N | 指定字体的颜色，支持：  red（红），black（黑），blue（蓝），purple（紫），grey（灰），brown（棕），tan(褐色)，cyan(青色) |

**示例：**

**{**

**"account"**: "mlimd@163.com"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"data"**: **{ },**

**"cost": 233,**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 241207**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "user not exist"

**}**

### 2.2、上传用户签名/印章图片

说明：

1. 为某一用户上传用户签名图片、或印章图片。
2. 印章图片的尺寸请按照实际尺寸（PS编辑图片时以厘米为单位的尺寸）设置，比如印章是“5厘米×3.5厘米”大小的，印章图片就设置为“5厘米×3.5厘米”，印章图片的dpi建议使用96或120，dpi太低签完后的清晰度不够，dpi太高则图片体积太大。
3. 上上签不对上传的图片做图像处理（如背景透明化等），请在上传前将印章图片处理好。

Request Path：**[server-host]**/signatureImage/user/upload/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 上传哪个用户的签名/印章图片就填该用户的唯一标识 |
| **imageData** | 图片文件内容 | Y |  | 图片经Base64编码后的字符串 |
| **imageName** | 签名/印章图片名称 | N | 50 | 传空或default表示更新默认的签名/印章图片。  企业用户如果有多个印章，可以指定印章名称，签署时用指定的印章名称 |

**示例：**

**{**

**"account"**: "abc@163.com"**,**

**"imageData"**: "图片经base64过的字符串"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"data"**: **{ },**

**"cost": 233,**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 241207**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "user not exist"

**}**

### 2.3、下载用户签名/印章图片

说明：下载用户的默认签名图片、印章图片数据，便于核对。

Request Path：**[server-host]**/signatureImage/user/download/

Method：GET

其他url参数：account，imageName

需要参与签名的其他URL参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 下载哪个用户的签名/印章图片就填该用户的唯一标识 |
| **imageName** | 签名/印章图片名称 | Y | 50 | 空字符串和default都表示默认的签名/印章图片。  企业用户如需下载自定义印章则此处填写上传时指定的印章名称 |

**请求参数（REQUEST BODY）：无**

**示例：**

**GET [**host**]/**signatureImage**/**user**/**download**/?**account**=**72dun32d&developerId**=**d445f5432**&**imageName=**&**rtick**=**17364899234**&**sign**=**a334df21323**&**signType**=**rsa

**响应（RESPONSE）：**

正常返回的response内容是签名图片的二进制数据（文件流），获取后自行存储。

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

## 3、文件存储服务

### 3.1、上传合同文件

说明：

1. 此接口是将文件保存到上上签，暂时只支持上传PDF文件、WORD、EXCEL、PPT文件。
2. 上传成功后得到一个文件编号，后续流程可以直接使用此文件编号发起合同。

Request Path：**[server-host]**/storage/upload/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 必须要指定一个用户唯一标识作为操作者 |
| **fdata** | 文件数据，base64编码 | Y |  | 例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fdata =Base64.encodeBase64String(bdata); |
| **fmd5** | 文件md5值 | Y |  | 例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fmd5 = DigestUtils.md5Hex(bdata); |
| **ftype** | 文件类型 | Y | 10 | 如PDF, docx, png等 |
| **fname** | 文件名 | Y | 50 | 文件名必须带上后缀名，例如“XXXX.pdf”、“XXXX.docx” |
| **fpages** | 文件页数 | Y | 3 |  |

**示例：**

**{**

**"account"**: "u-1112343242"**,**

**"fmd5"**: "3424324112343242"**,**

**"ftype"**: "PDF"**,**

**"fname"**: "contract.PDF"**,**

**"fpages"**: "7"**,**

**"fdata"**: "............."

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **fid** | 文件编号 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"fid"**: "f1343423222"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 3.3、为PDF文件添加元素

说明：为PDF文件添加元素，目前支持的元素类型为text、image。添加元素后会得到一个新的文件编号，此文件编号可以用于后续流程发起合同。

Request Path：**[server-host]**/storage/addPDFElements/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 必须要指定一个用户唯一标识作为操作者 |
| **fid** | 源文件编号 | Y |  | 源文件必须是PDF文件格式 |
| **elements** | 要添加的元素集合 | Y | 10 | json array格式 |

**示例：**

**{**

**"account"**: "u-1112343242"**,**

**"fid"**: "f1343423222"**,**

**"elements"**: **[**

**{**

**"pageNum"**: "新添加元素所在的页码"**,**

**"x"**: "新添加元素x坐标, 用百分比表示，取值0.0~1.0"**,**

**"y"**: "新添加元素y坐标, 用百分比表示，取值0.0~1.0"**,**

**"type"**: "新添加元素的类型. 目前支持：text, image两种"**,**

**"value"**: "新添加元素的内容. 如果是text类型，为文本;如果是image类型，为base64编码后的图片内容"**,**

**"fontSize"**: "如果新添加元素是text类型，可以用来指定新添加元素的字体大小。默认14"

**}**

**]**

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **fid** | 产生新的文件编号 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"fid"**: "f1343423222"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 3.4、下载文件

说明：根据文件编号（fid）下载指定文件。可用于查看添加完元素的PDF文件是否符合预期效果。

Request Path：**[server-host]**/storage/download/

Method：GET

其他url参数：无

需要参与签名的其他URL参数：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **fid** | 文件编号 | Y | 20 | 上传合同文件得到的文件编号 |

**请求参数（REQUEST BODY）：**无

**示例：**

**GET [**server-host**]/**storage**/**download**/?** developerId**=**d445f5432**&**rtick**=**17364899234**&**sign**=**a2334df21323**&**signType**=**rsa**&**fid=150496133501000001

**响应（RESPONSE）：**

RESPONSE DATA参数如下：文件流（byte[]）

正常示例：

正常则返回文件流，开发者自行另存到指定的路径。

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 3.5、PDF文件验签

说明：

上传PDF文件验证签名，返回PDF文件是否有证书签名、签名/文件在签名后是否有篡改，以及每个证书签名的具体信息。

Request Path：**[server-host]**/pdf/verifySignatures/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **pdfData** | PDF文件 | Y |  | PDF文件base64编码过的字符串 |

**示例：**

{

**"pdfData"**: "ZHNmYXNnZHNnZHNhZ3Nn.."

}

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** |  | | |
| **verifyResult** | 验证结果 | 0-文档无证书签名  1-文档未被篡改  2-文档被篡改 | | |
| **signatureDetails** | 签名明细，json array格式 | 参数 | 说明 | |
| signatureNumber | | 签名序号（ADOBE PDF阅读器查看的修订版序号） |
| dn | | 证书DN项 |
| signTime | | 签署时间 |
| serialNumber | | 证书序列号 |
| singer | | 签署者名称 |
| signIsModified | | 该签名有无被篡改：  0-未篡改 1-已篡改 |
| documentIsModified | | 签名后文档有无被篡改：  0-未篡改 1-已篡改 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"verifyResult"**: "1"**,**

**"signatureDetails"**: **[**

**{**

**"signatureNumber"**: 1**,**

**"singer"**: "测试企业有限公司"**,**

**"serialNumber"**: "1003623585"**,**

**"dn"**: "C=CN,O=Zhejiang Digital Certificate Authority,CN=ZJCA OCA2"**,**

**"signTime"**: "2017-12-22 15:54:26"**,**

**"signIsModified"**: "0"**,**

**"documentIsModified"**: "0"

**},**

**{**

**"signatureNumber"**: "3"**,**

**"serialNumber"**: "74c8003d0001d901"**,**

**"singer"**: "测试 \_青岛企业"**,**

**"signTime"**: "2017-12-22 15:56:35"**,**

**"signIsModified"**: "0"**,**

**"dn"**: "C=CN,O=Zhejiang Digital Certificate Authority,CN=ZJCA OCA3"**,**

**"documentIsModified"**: "0"

**}**

**]**

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

## 4、单文档合同服务

* 关于toB/toC合同：合同签署人中除了发送者自己外还有至少一个企业就是toB合同，否则就是toC合同。

### 4.1、上传并创建合同

说明：

* 此接口是将“3.1、上传合同文件”与“4.11、创建合同”的封装在一起，方便开发者调用。调用此接口后，将文件保存到上上签，暂时只支持上传PDF文件。上传成功后得到一个合同编号，后续流程可以直接使用此合同编号进行签署。
* 如果开发者需要用到空白合同当模版，添加不同的PDF元素时，请分开使用“3.1、上传合同文件”、“3.2、PDF文件添加元素”与“4.11、创建合同”。

Request Path：**[server-host]**/storage/contract/upload/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y | 64 | 必须要指定一个用户唯一标识作为操作者 |
| **fmd5** | 文件md5值 | Y | 100 | 例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fmd5 = DigestUtils.md5Hex(bdata); |
| **ftype** | 文件类型 | Y | 10 | 仅支持PDF |
| **fname** | 文件名 | Y | 50 | 文件名必须带上后缀名，例如“XXXX.pdf” |
| **fpages** | 文件页数 | Y | 3 |  |
| **fdata** | 文件数据，base64编码 | Y | 5120000 | 例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fdata =Base64.encodeBase64String(bdata); |
| **title** | 合同标题 | Y | 50 | 客户可以将自己的业务合同编号、或合同标题放此处 |
| **expireTime** | 合同能够签署的截止时间 | Y | 20 | 合同必须在指定的截止时间之前完成签署，一旦过期则此合同将无法被签署。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”； |
| **description** | 合同描述 | N | 200 |  |
| **hotStoragePeriod** | 热存周期 | N |  | 此参数是合同文件在热存储中保留的时间长度，单位为秒。保存在热存储中的合同数据，自合同结束时间算起，超过此此参数设定时长的合同文件，会转移到冷存储中。计算示例：如保存365天，则值为3600\*24\*365=31536000。参数可为空，为空时默认值为1年（31536000）。取值范围为3600（1小时）~157680000（5年） |

**示例：**

**{**

"account"**:** "mlimd@163.com"**,**

"fmd5"**:** "文件的md5值"**,**

"ftype"**:** "pdf"**,**

"fname"**:** "滴滴电子发票.pdf"**,**

"fpages"**:** "1"**,**

"fdata"**:** "文件的base64字符串"**,**

"title"**:** "测试合同标题"**,**

"description"**:** "测试合同描述"**,**

"expireTime"**:** "1531242131"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **contractId** | 创建的合同ID，在后续接口中使用 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"contractId"**: "150993706101000005"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.2、签署合同（即自动签）

说明：签署合同，可直接用于无感知的快捷签署。此接口要求签署者必须拥有自己的数字证书。可以无需先调用“添加签署者”接口与“设置合同签署参数”接口。

Request Path：**[server-host]**/contract/sign/cert/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | |
| **contractId** | 合同编号 | Y |  |  | | |
| **signer** | 签署者 | Y |  | 即签署者的account | | |
| **signaturePositions** | 指定的签署位置，json array格式 | Y |  | 每一项为一个json对象 | | |
|  |  | 最大长度 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| y | 纵坐标，同上 | 10 |
| pageNum | 页码。1开始，不能超过实际页数 | 3 |
| rptPageNums | 当前位置的签名需要复制到的目标页码列表。该参数用于控制是否将当前位置的签名复制到其他页。参数启用之后，当前页的签名会同时出现在其他页的相同位置。参数值格式如下： 多个页码使用英文逗号(,)分隔。 参数不存在或者参数值为空表示不复制， 参数值为0表示复制到所有页， 参数值为其他表示复制到指定页。 参数值示例："0" "1,2,3,4,5,6" "1,3,5,7,9" |  |
| **signatureImageName** | 签名图片名称 | N |  | 企业用户可以指定上传过的印章图片中的某一张作为本次签署的印章图片，取“上传用户签名/印章图片”接口设置的**imageName**的值 | | |
| **signatureImageData** | 签名图片 | N |  | 用户指定的签名图片，Base64字符串。允许为空，为空时使用用户默认的签名图片。  使用自动签的用户，可以越过“2.1、创建用户签名/印章图片”接口、或“2.2、上传用户签名/印章图片”接口，直接在此提供签名/印章图片即可。 | | |
| **signatureImageWidth** | 签名图片显示宽度 | N |  | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。 如果有一个为空，则按照同时为空处理，即使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。 如果同时有值，则图片按照两个宽高值拉伸填充，效果可能会产生变形。 | | |
| **signatureImageHeight** | 签名图片显示高度 | N |  | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。 如果有一个为空，则按照同时为空处理，即使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。 如果同时有值，则图片按照两个宽高值拉伸填充，效果可能会产生变形。 | | |

**示例：**

**{**

**"contractId"**: "cU34344234"**,**

**"signer"**: "u-1112343242"**,**

**"signatureImageData"**: "cU34344234"**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.05"**,**

**"y"**: "0.71"**,**

**"rptPageNums"**: "2,3,4,5,6,7"

**},**

**{**

**"pageNum"**: "2"**,**

**"x"**: "0.05"**,**

**"y"**: "0.81"

**}**

**]**

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.3、发送合同（即手动签，指定图片大小）

说明：获取合同手动签署的页面URL，将此URL发送给指定用户，即可在此页面进行手动签操作完成签署。增加了参数“**signatureImageWidth**”与“**signatureImageHeight**”，可以指定签名图片显示的宽高。

Request Path：[**server-host**]/contract/send/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | | | **是否必填** | **最大长度** | **默认值** |
| **contractId** | 合同编号 | | | Y |  |  |
| **signer** | 指定给哪个用户看 | | | Y |  |  |
| **dpi** | 预览图片清晰度，枚举值：96-低清（默认），120-普清，160-高清，240-超清 | | | N | 3 | 96 |
| **signaturePositions** | **参数名** | **说明** | **最大长度** | Y |  |  |
| y | 纵坐标，同上 | 10 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| pageNum | 页码，从1开始，不能超过合同最大页数 | 3 |
| rptPageNums | 当前位置的签名需要复制到的目标页码列表。该参数用于控制是否将当前位置的签名复制到其他页。参数启用之后，当前页的签名会同时出现在其他页的相同位置。参数值格式如下： 多个页码使用英文逗号(,)分隔。 参数不存在或者参数值为空表示不复制， 参数值为0表示复制到所有页， 参数值为其他表示复制到指定页。 参数值示例："0" "1,2,3,4,5,6" "1,3,5,7,9" |  |
| type | 日期类型专用，当签署位置是“日期”类型的签名时，本参数需要填写“date” |  |
| dateTimeFormat | 日期类型专用，当签署位置是“日期”类型的签名时，本参数需要填写日期的格式，格式支持以下类型：  MM-dd-yyyy  yyyy-MM-dd  MM.dd.yyyy  Yyyy.MM.dd  MM/dd/yyyy  Yyyy/MM/dd |  |
| fontSize | 日期类型专用，当签署位置是“日期”类型的签名时，本参数需要填写字号大小 |  |
| **isAllowChangeSignaturePosition** | 在有指定signaturePOSTion参数的情况下，是否允许拖动签名位置。取值1/0。（0：不允许，1：允许） | | | N | 1 | 0 |
| **expireTime** | 签署链接的到期时间，unix时间戳格式（秒数）。超过此时间则无法打开链接页面，需要获取新的签署合同链接。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”。  注：并不是合同的可签署到期时间，只是此签署链接的有效期。  由于返回的是短链接，如果没有设置则默认是7天，如果设置了以设置的为准，但有效期最大不能超过7天。 | | | N | 20 | 0 |
| **returnUrl** | 手动签署时，当用户签署完成后，签署页面指定回跳的页面地址，如果没填此参数则签署完成后跳转到合同预览页面。  如果returnurl带有“http”或“https”开头则判定h5跳转，如果returnurl不带“http”或“https”开头则判定为微信小程序跳转。 | | | N |  |  |
| **vcodeMobile** | 手动签署时指定接收手机验证码的手机号，如果需要采用手动签署页面校验短信验证码，则此项必填。  如果手动签不需要短信校验，则此项可不填。  自动签可不填此项。 | | | N |  | 20 |
| **isDrawSignatureImage** | 手动签署时是否手绘签名。取值0/1/2。  0点击签名图片不会触发手写面板（禁止手写）。  1点击签名图片能触发手写面板（既可手写也可使用默认签名）。  2强制必须手绘签名（只能手写不允许使用默认签名），初始不显示用户签名图片，只显示实时手写的签名。 | | | N | 1 | 1 |
| **signatureImageName** | 签名/印章图片。取值default或者指定的印章名称。（手动签只有参数version=3时才支持企业多印章，填写自定义的印章名称） | | | N | default | 50 |
| **sid** | 开发者自定义的签署流水号（比如第几次签署之类），如有填写，则在异步推送签署结果时包含此签署流水号 | | | N | 30 |  |
| **pushUrl** | 此处有配置则签署推送消息优先使用此配置，如果此处没有配置，则取开发者统一配置的异步推送地址 | | | N |  |  |
| **version** | 手动签的版本，默认为2，枚举值：  2-v2版本,  3-v3版本，UI优化，移动端手绘面板自动横屏 | | | N | 1 | 2 |
| **readAll** | 枚举值0/1，默认为0  0-无需拖动到页面底部即可确认签署  1-必须拖动到页面底部，表示阅读完毕才能确认签署 | | | N |  |  |
| **app** | app 人脸比对签署（即app内刷脸签署）用到的app参数，参数内容为JSBridge，即手动签页面将此参数传递给客户app以便客户app调用对应的SDK。 一般情况，如果不需要用户跳出手动签页面执行其他app操作的话，则REQUEST BODY不需要带有此参数。  新版手动签在返回签署人配置的时候，会将此参数返回给前端，前端需要将此参数传递给app。 | | | N |  |  |
| **signatureImageWidth** | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | | | N |  |  |
| **signatureImageHeight** | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | | | N |  |  |
| **isShowHandwrittenTime** | 枚举值0/1，默认为0   1. 手写面板及签名图片上显示手写签名操作的时间   0-不显示该时间 | | | N | 1 | 0 |
| **isFaceAuth** | 是否使用刷脸签。   1. 刷脸签，使用刷脸校验；   0或其他-非刷脸签，使用短信校验或无校验； | | | N |  |  |
| **signMethod** | 签署方式，参数值解释如下：  qrc：返回合同签署的二维码；  url：返回合同签署的url。  当signMethod不填写时，返回合同签署的url | | | N |  |  |

**示例：**

**{**

**"contractId"**: "cU34344234"**,**

**"expireTime"**: "0"**,**

**"dpi"**: "120"**,**

**"signer"**: ""**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.4622"**,**

**"y"**: "0.5852999973297119"**,**

**"rptPageNums"**: ""**,**

**"type"**: "date"**,**

**"dateTimeFormat"**: "MM-dd-yyyy"**,**

**"fontSize"**: "18"

**},**

**{**

**"pageNum"**: "6"**,**

**"x"**: "0.4622"**,**

**"y"**: "0.7852999973297119"

**}**

**],**

**"isAllowChangeSignaturePosition"**: "0"**,**

**"returnUrl"**: ""**,**

**"vcodeMobile"**: ""**,**

**"isDrawSignatureImage"**: "1"**,**

**"signatureImageName"**: "default"**,**

**"pushUrl"**: "https://xxxxxxx"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

手动签署页面的url，签署者在此页面上进行签署操作。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"url"**: "https://..........."

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.4、撤销合同

说明：当发现合同内容出错等情况时，可用来直接取消未结束的合同（由开发者发起的取消，不需要确认签署者），避免造成错误的合同完成了签署。

Request Path：[**server-host**]/contract/cancel/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **contractId** | 合同编号 | Y |  |

**示例：**

**{**

**"contractId"**: "1703466073594003458"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.5、锁定并结束合同

说明：当所有签署者都签署完毕后，使用此接口添加上上签系统的锁定签名，并且结束整个合同签署过程，将合同状态更新为“已完成”，不允许任何人再进行签署。只有结束的合同才能提供公证服务。

Request Path：**[server-host]**/storage/contract/lock/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **contractId** | 合同编号 | Y |  |

**示例：**

**{**

"contractId"**:** "1703465339807805447"

**}**

**响应（RESPONSE）：**errno返回0则为正常，errno返回错误码则为异常，此时观察具体的errmsg具体分析。errno与errmsg请参阅本文《四、附录：返回信息编号列表》，返回信息编号列表持续更新。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.6、查询合同信息

说明：获取合同的详细信息，如签署者信息、合同状态等。

特殊应用：本接口支持跨开发者的查询。场景是：开发者A（合同的发起者）登录上上签开放平台设置“开发者授权”，给开发者B（需要查询的其他开发者），开发者B就能使用此接口查询A指定的合同信息。开发者授权的操作如下截图所示：



Request Path：[**server-host**]/contract/getInfo/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **contractId** | 合同编号 | Y |  |

**示例：**

**{**

"contractId"**:** "cU34344234"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |  |
| --- | --- | --- |
| **参数名** | **说明** |  |
| **contractId** | 合同编号 |  |
| **fid** | 文件编号 |  |
| **userId** | 用户id | 即创建者的account在上上签系统内的用户id |
| **senderAccount** | 合同创建者唯一标识（之前版本中的**senderUserId**作废） | 即创建者的account |
| **description** | 合同描述 |  |
| **expireTime** | 合同签署的到期时间 | Unix时间戳（秒数），格式：1490976000 |
| **title** | 标题 |  |
| **sendTime** | 合同的发送时间 | 添加第一个签署者的时间，当第一个签署者添加成功即认为合同已发送 |
| **finishTime** | 合同的完成时间 | 调用锁定合同接口之后结束合同的时间 |
| **signers** | 合同签署者account集合. json array形式。 | "signers": [  "mlimd@163.com"  ] |
| **developerId** | 开发者编号 |  |
| **pages** | 合同页数 |  |
| **status** | 合同状态 | 2：已创建；3：已发送，正在签署中；4：拒绝签署,已取消；5：已完成；9：已过期； |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"fid"**: "1702695409383636995"**,**

**"senderAccount"**: "1364808"**,**

**"userId"**: "1693391393705164808"**,**

**"description"**: "postman测试合同"**,**

**"expireTime"**: "1490976000"**,**

**"title"**: "测试"**,**

**"sendTime"**: "2017-04-10 17:31:46"**,**

**"finishTime"**: "2017-04-10 17:31:46"**,**

**"signers"**: **[**

"mlimd@163.com"

**],**

**"developerId"**: "1678748413568483333"**,**

**"pages"**: "1"**,**

**"contractId"**: "1702698069495119878"**,**

**"status"**: "3"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.7、获取合同签署参数

说明：获取已经设置过的签署者合同签署参数。

Request Path：[**server-host**]/contract/getSignerConfig/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **默认值** |
| **contractId** | 合同编号 | Y |  |  |
| **account** | 签署者 | Y |  |  |

**示例：**

**{**

"contractId"**:** "cU34344234"**,**

"account"**:** "2348@qq.com"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **signaturePositions** | 默认的签署位置 |
| **isAllowChangeSignaturePosition** | 在有指定signaturePOSTion参数的情况下，是否允许拖动签名位置。取值1/0。（0：不允许，1：允许，下面都是这样的） |
| **password** | 合同密码 |
| **isVerifySigner** | 手动签署时是否验证account身份。取值0/1。如果要验证account身份，则忽略"查看密码"和"验证码"的设置 |
| **vcodeMail** | 手动签署时指定接收邮件验证码的邮箱。 |
| **vcodeMobile** | 手动签署时指定接收手机验证码的手机号。 |
| **isDrawSignatureImage** | 手动签署时是否手绘签名。取值0/1 |
| **signatureImageName** | 签名/印章图片。取值default或者指定的印章名称。 |
| certType | 证书类型 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.05"**,**

**"y"**: "0.01"

**},**

**{**

**"pageNum"**: "2"**,**

**"x"**: "0.05"**,**

**"y"**: "0.01"

**}**

**],**

**"isAllowChangeSignaturePosition"**: "0"**,**

**"pushUrl"**: ""**,**

**"password"**: ""**,**

**"isVerifySigner"**: "0"**,**

**"vcodeMail"**: ""**,**

**"vcodeMobile"**: ""**,**

**"isDrawSignatureImage"**: "1"**,**

**"signatureImageName"**: "default"**,**

**"certType"**: ""

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.8、查询合同签署者状态

说明：查询合同签署者的签署状态。

Request Path：[**server-host**]/contract/getSignerStatus/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **contractId** | 合同编号 | Y |  |

**示例：**

**{**

"contractId"**:** "cU34344234"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

data内容为合同签署者状态，为json格式，格式为“account - 状态码”成对出现。状态码：1 － 未签署; 2 - 已签署。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"mlimd@163.com"**: "2"**,**

**"13067825582"**: "2"**,**

**"13867410069"**: "2"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.9、下载合同文件

说明：通过合同编号下载对应的合同文件。

Request Path：[server-host]/contract/download/

Method：GET

其他url参数：**contractId**

URL示例：https://[server-host]/contract/download/?developerId=xxxxxx&rtick=xxxxxx&signType=rsa&sign=xxxxxx&contractId=xxxxx

需要参与签名的其他URL参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **contractId** | 合同编号 | Y |  |

**请求参数（REQUEST BODY）：**无

**示例：**

**GET [**server-host**]/**storage**/**contract**/**download**/?** developerId**=**d445f5432**&**rtick**=**17364899234**&**sign**=**a5334df21323**&**signType**=**rsa**&**contractId=150496133501000001

**响应（RESPONSE）：**正常则返回pdf文件的文件流，errno返回错误码则为异常，此时观察具体的errmsg具体分析。errno与errmsg请参阅本文《四、附录：返回信息编号列表》，返回信息编号列表持续更新。

正常示例：

**文件流**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.10、获取预览页URL

说明：获取合同预览的页面URL，在此页面预览合同内容。

Request Path：[**server-host**]/contract/getPreviewURL/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **默认值** |
| **contractId** | 合同编号 | Y |  |  |
| **account** | 指定给哪个用户看 | Y |  |  |
| **dpi** | 预览图片清晰度，枚举值：96-低清（默认），120-普清，160-高清，240-超清 | N | 3 | 96 |
| **expireTime** | 预览链接的过期时间，unix时间戳。超过此时间则无法打开预览链接，需要获取新的预览链接。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”。  注：并不是合同的可签署到期时间，只是此预览链接的有效期。  如果没有设置则默认是7天，如果设置了以设置的为准，但有效期最大不能超过7天。 | N | 20 | 0 |

**示例：**

**{**

"contractId"**:** "cU34344234"**,**

"expireTime"**:** "0"**,**

"dpi"**:** "96"**,**

"account"**:** ""

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

预览合同页面的url，签署者在此页面上预览合同，但不可操作签署。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"url"**: "https://..........."

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.11、创建合同

说明：使用已得到的文件编号创建合同，得到合同编号，此合同还未添加签署者，属于草稿性质。每份合同必须要有创建者account。

Request Path：[**server-host**]/contract/create/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 必须要指定一个用户唯一标识作为合同的创建者（建议统一为开发者的account，或者需要进行分类统计的某些account，以便后期按此account进行统计） |
| **fid** | 文件编号 | Y |  | 上传合同文件得到的文件编号 |
| **expireTime** | 合同能够签署的截止时间 | Y |  | 合同必须在指定的截止时间之前完成签署，一旦过期则此合同将无法被签署。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”； |
| **title** | 合同标题 | Y | 50 | 客户可以将自己的业务合同编号、或合同标题放此处 |
| **description** | 合同描述 | N | 200 |  |
| **hotStoragePeriod** | 热存周期 | N |  | 此参数是合同文件在热存储中保留的时间长度，单位为秒。保存在热存储中的合同数据，自合同结束时间算起，超过此此参数设定时长的合同文件，会转移到冷存储中。计算示例：如保存365天，则值为3600\*24\*365=31536000。参数可为空，为空时默认值为1年（31536000）。取值范围为3600（1小时）~157680000（5年） |

**示例：**

**{**

**"account"**: "mlimd@163.com"**,**

**"fid"**: "1702695409383636995"**,**

**"expireTime"**: "1513311132"**,**

**"title"**: "测试"**,**

**"description"**: "postman测试合同"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **contractId** | 合同编号 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"contractId"**: "1694593128888888"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 4.21、模版签署功能

模版功能包含了一整套从发到签的接口，使用模版功能，可以实现“编辑一次、重复使用”的目的，并且无需计算签署位置坐标等参数，为开发者对接调试提供极大方便。

此功能与开放平台的页面功能“模版管理”相结合，使用流程为：

在开放平台的“模版管理”页面上可以可视化的创建、编辑模版，保存后，可以通过下列接口进行合同的创建、签署。

在“模版管理”页面上可以查看每一份模版的变量详情，这些变量详情中有一些内容会在“创建合同”、“签署合同”等不同的环节用到。

|  |  |  |  |
| --- | --- | --- | --- |
| **变量类型（type）** | **说明** | **用途** | **何时会用到** |
| 50 | 盖章 | 企业用户盖章的位置 | 签署 |
| 40 | 签名 | 个人用户签名的位置 | 签署 |
| 20 | 签署日期 | 具体每个用户可能用到的签署日期 | 签署 |
| 11 | 文本 | 自定义的合同变量，比如在模版上要填充的“地址”、“身份证号”等等 | 创建合同 |

变量详情查看的具体内容如下所示：

**{**

**"fid"**: "7388583176326841500"**,**

**"templateVars"**: **[**

**{**

**"isRequired"**: "1"**,**

**"overFlow"**: "0"**,**

**"type"**: "50"**,** //表示“盖章”

**"pageNum"**: "1"**,** //签署位置的页码

**"fontFamily"**: ""**,**

**"dateTimeFormat"**: "yyyy-MM-dd HH:mm:ss"**,**

**"name"**: "甲方盖章1"**,** //调签署接口时用到name变量

**"tagType"**: "0"**,**

**"width"**: "0.2052896725440806"**,**

**"x"**: "0.17758186397984888"**,** //签署位置的x坐标

**"y"**: "0.182546749777382"**,** //签署位置的y坐标

**"fontSize"**: "0"**,**

**"tag"**: "盖章"**,**

**"text"**: ""**,**

**"height"**: "0.12733748886910062"

**},**

**{**

**"isRequired"**: "1"**,**

**"overFlow"**: "0"**,**

**"type"**: "40"**,** //表示“签名”

**"pageNum"**: "1"**,** //签署位置的页码

**"fontFamily"**: ""**,**

**"dateTimeFormat"**: "yyyy-MM-dd HH:mm:ss"**,**

**"name"**: "乙方签名1"**,** //调签署接口时用到此变量

**"tagType"**: "0"**,**

**"width"**: "0.0654911838790932"**,**

**"x"**: "0.5478589420654912"**,** //签署位置的x坐标

**"y"**: "0.25645592163846836"**,** //签署位置的y坐标

**"fontSize"**: "0"**,**

**"tag"**: "签名"**,**

**"text"**: ""**,**

**"height"**: "0.030276046304541407"

**},**

**{**

**"isRequired"**: "1"**,**

**"overFlow"**: "0"**,**

**"type"**: "20"**,** //表示“签署日期”

**"pageNum"**: "1"**,** //签署位置的页码

**"fontFamily"**: ""**,**

**"dateTimeFormat"**: "yyyy-MM-dd HH:mm:ss"**,**

**"name"**: "甲方签署日期1"**,** //调签署接口时用到此变量

**"tagType"**: "0"**,**

**"width"**: "0.2506297229219144"**,**

**"x"**: "0.25440806045340053"**,** //签署位置的x坐标

**"y"**: "0.3873552983081033"**,** //签署位置的y坐标

**"fontSize"**: "16"**,**

**"tag"**: "签署日期"**,**

**"text"**: ""**,**

**"height"**: "0.02315227070347284"

**},**

**{**

**"isRequired"**: "1"**,**

**"overFlow"**: "0"**,**

**"type"**: "11"**,** //表示“自定义文本”

**"pageNum"**: "1"**,**

**"fontFamily"**: "SimSun"**,**

**"dateTimeFormat"**: "yyyy-MM-dd HH:mm:ss"**,**

**"name"**: "甲方地址1"**,** //在创建合同时会用到

**"tagType"**: "0"**,**

**"width"**: "0.12594458438287154"**,**

**"x"**: "0.6964735516372796"**,**

**"y"**: "0.3846838824577026"**,**

**"fontSize"**: "18.7"**,**

**"tag"**: "文本"**,**

**"text"**: ""**,**

**"height"**: "0.02226179875333927"**,**

**"warp"**: "true"

**},**

**{**

**"isRequired"**: "1"**,**

**"overFlow"**: "0"**,**

**"type"**: "11"**,**

**"pageNum"**: "1"**,**

**"fontFamily"**: "SimSun"**,**

**"dateTimeFormat"**: "yyyy-MM-dd HH:mm:ss"**,**

**"name"**: "乙方身份证"**,**

**"tagType"**: "0"**,**

**"width"**: "0.12594458438287154"**,**

**"x"**: "0.7670025188916877"**,**

**"y"**: "0.4790739091718611"**,**

**"fontSize"**: "18.7"**,**

**"tag"**: "文本"**,**

**"text"**: ""**,**

**"height"**: "0.02226179875333927"**,**

**"warp"**: "true"

**}**

**],**

**"pageCount"**: 2**,**

**"fileName"**: "测试合同.pdf"**,**

**"developerId"**: "1678748413568483333"**,**

**"description"**: "未描述"**,**

**"titleHash"**: "011c945f30ce2cbafc452f39840f025693339c42"**,**

**"title"**: "1111"**,**

**"isDel"**: "0"**,**

**"tid"**: "151393625101000001" //模版编号

**}**

#### 4.21.0、获取模版变量

说明：使用此接口获取模版的所有变量，用于生成合同文件或签署合同。

Request Path：**[server-host]**/template/getTemplateVars/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **tid** | 模版编号 | Y |  |
| **isRetrieveAllVars** | 是否获取所有变量 | Y | 0只返回变量的type和name；  1返回变量的所有字段； |

**示例：**

**{**

**"tid"**: "111222333"**,**

**"isRetrieveAllVars"**: "0"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **templateVars** | 模版变量, jsonArray格式 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"templateVars"**: **[**

**{**

**"name"**: "甲方住址1"**,**

**"type"**: "11"

**},**

**{**

**"name"**: "签署日期1"**,**

**"type"**: "20"

**},**

**{**

**"name"**: "甲方签名1"**,**

**"type"**: "40"

**},**

**{**

**"name"**: "乙方盖章1"**,**

**"type"**: "50"

**}** **]**

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 4.21.1、通过模版生成合同文件

说明：

在开放平台的“开发者模版”页面编辑保存好模版之后，就可以通过接口使用该模版。创建合同前先调本接口通过模版功能创建PDF文件。

Request Path：**[server-host]**/template/createContractPdf/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 合同创建者账号 | Y |  |  |
| **tid** | 模版编号 | Y |  | “开发者模版”页面模版保存后产生模版编号，可从页面上复制 |
| **templateValues** | 模版变量，jsonObject格式 | Y |  | 将模版变量中需要填充的内容（即变量值）在此提交。  由于在编辑模版时已经拖放了位置、并且已经保存在上上签系统中，因此这里无需再提交位置参数，只需要传入变量名称与对应的值即可。  模版变量可以在开发者模版管理页面获取。 |
| **groupValues** | 模版变量组，jsonObject格式 | N |  | 模版变量组名，每个变量可以设置归什么组，然后用这个组名来传递变量。  **templateValues和groupValues不能全部为空。** |

**示例：**

**{**

**"account"**: "111222333@qq.com"**,**

**"tid"**: "1"**,**

**"templateValues"**: **{**

**"甲方姓名"**: "张三"**,**

**"甲方身份证号"**: "330101198111110311"**,**

**"甲方地址"**: "浙江省杭州市西湖区灵隐路1号"**,**

**"乙方企业名称"**: "北京朝阳群众信息技术有限公司"**,**

**"乙方证件号码"**: "330101012345678"**,**

**"乙方地址"**: "北京市朝阳区朝阳大厦1024室"

**},**

窗体顶端

**"groupValues"**: **{**

**"group1"**: "aaa"**,**

**"group2"**: "bbb"

**}**

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **templateToken** | 通过模版创建合同需要这个token，格式是一个Base64编码字符串 ＋ "." + 一个自签名串 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"templateToken "**: "一串字符串"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 4.21.2、通过模版创建合同

说明：

在“通过模版生成合同文件”得到**templateToken**之后，再调用本接口创建合同，得到合同编号。后续用模版变量签署必须使用本接口返回的合同编号。

Request Path：**[server-host]**/contract/createByTemplate/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 必须要指定一个用户唯一标识作为合同的创建者（建议统一为开发者的account，或者需要进行分类统计的某些account，以便后期按此account进行统计） |
| **tid** | 模版编号 | Y |  | 要用来创建合同的模版编号 |
| **templateToken** | templateToken | Y |  | 上一个接口/template/createContractPdf/返回的templateToken |
| **title** | 合同标题 | Y | 50 | 客户可以将自己的业务合同编号、或合同标题放此处 |
| **description** | 合同描述 | N | 200 |  |
| **expireTime** | 合同能够签署的到期时间 | N |  | 合同必须在指定的到期时间之前完成签署，一旦过期则此合同将无法被签署。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”；如果不填则默认为创建后加7天有效期。 |
| **hotStoragePeriod** | 热存周期 | N |  | 此参数是合同文件在热存储中保留的时间长度，单位为秒。保存在热存储中的合同数据，自合同结束时间算起，超过此此参数设定时长的合同文件，会转移到冷存储中。计算示例：如保存365天，则值为3600\*24\*365=31536000。参数可为空，为空时默认值为1年（31536000）。取值范围为3600（1小时）~157680000（5年） |

**示例：**

**{**

**"account"**: "mlimd@163.com"**,**

**"tid"**: "1702695409383636995"**,**

**"templateToken"**: "上一个接口返回的一串字符串"**,**

**"expireTime"**: "1513311132"**,**

**"title"**: "测试"**,**

**"description"**: "postman测试合同"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **contractId** | 合同编号 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"contractId"**: "150993706101000005"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 4.21.3、用模版变量签署合同

说明：使用模版功能创建合同后，使用本接口完成签署。

Request Path：**[server-host]**/contract/sign/template/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **contractId** | 合同编号 | Y |  |  |
| **tid** | 模版编号 | Y |  |  |
| **vars** | 模版变量值 | Y | JSONObject格式 | 每一项为一个jsonObject，格式类似如下：  **varName——**模版变量名，比如“甲方盖章”、“甲方签署日期”等   |  |  | | --- | --- | | **参数** | **说明** | | **account** | 签署者账号，同一份合同上的同一个签署者不可分多次签署。不同的签署者分次签署。 | | **signatureImageData** | 签名图片，base64编码字符串。   * 如果事先调用上传签名图片接口为签署者上传过签名图片，则此处可以不传，如果传了则优先使用此处上传的签名图片。 * 如果变量是签署日期类型则此项不传，传了会报错。 | |
| **loginIp** | 登录IP | N |  | 签署者登录开发者平台的IP地址 |
| **signIp** | 签署IP | N |  | 签署者操作通过开发者平台操作签署时的IP |

**示例：**

**{**

**"contractId"**: ""**,**

**"vars"**: **{**

**"甲方盖章"**: **{**

**"account"**: ""**,**

**"signatureImageData"**: "base64编码"

**},**

**"甲方签署日期"**: **{**

**"account"**: ""

**}**

**},**

**"tid"**: ""**,**

**"signIp"**: ""**,**

**"loginIp"**: ""**,**

**"loginTime"**: ""

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：无

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{** **},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 4.21.4、获取模版信息

说明：使用此接口获取模版的所有信息，包括：文件编号、模版标题、模版变量等。

Request Path：**[server-host]**/template/getTemplate/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **tid** | 模版编号 | Y |  |

**示例：**

**{**

**"tid"**: "111222333"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **fid** | 模版文件编号 |
| **title** | 模版标题 |
| **description** | 模版描述 |
| **templateVars** | 模版变量, jsonArray格式 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost"**: 43**,**

**"data"**: **{**

**"template"**: **{**

**"fid"**: "876328732648726872"**,**

**"templateVars"**: **[**

**{**

**"isRequired"**: "1"**,**

**"overFlow"**: "0"**,**

**"type"**: "50"**,**

**"pageNum"**: "1"**,**

**"fontFamily"**: ""**,**

**"dateTimeFormat"**: "yyyy-MM-dd HH:mm:ss"**,**

**"name"**: "stamp"**,**

**"tagType"**: "0"**,**

**"width"**: "0.2055485498108449"**,**

**"x"**: "0.1450189155107188"**,**

**"y"**: "0.18788958147818344"**,**

**"fontSize"**: "18.7"**,**

**"tag"**: "盖章"**,**

**"maxLength"**: "20"**,**

**"height"**: "0.12733748886910062"**,**

**"warp"**: "true"

**},**

**{**

**"isRequired"**: "1"**,**

**"overFlow"**: "0"**,**

**"type"**: "40"**,**

**"pageNum"**: "2"**,**

**"fontFamily"**: ""**,**

**"dateTimeFormat"**: "yyyy-MM-dd HH:mm:ss"**,**

**"name"**: "sign"**,**

**"tagType"**: "0"**,**

**"width"**: "0.06557377049180328"**,**

**"x"**: "0.1437578814627995"**,**

**"y"**: "0.03829029385574354"**,**

**"fontSize"**: "18.7"**,**

**"tag"**: "签名"**,**

**"maxLength"**: "20"**,**

**"height"**: "0.030276046304541407"**,**

**"warp"**: "true"

**},**

**{**

**"isRequired"**: "1"**,**

**"overFlow"**: "0"**,**

**"type"**: "20"**,**

**"pageNum"**: "2"**,**

**"fontFamily"**: ""**,**

**"dateTimeFormat"**: "yyyy-MM-dd HH:mm:ss"**,**

**"name"**: "date"**,**

**"tagType"**: "0"**,**

**"width"**: "0.12610340479192939"**,**

**"x"**: "0.7288776796973518"**,**

**"y"**: "0.09884238646482636"**,**

**"fontSize"**: "18.7"**,**

**"tag"**: "签署日期"**,**

**"maxLength"**: "20"**,**

**"height"**: "0.02226179875333927"**,**

**"warp"**: "true"

**},**

**{**

**"isRequired"**: "1"**,**

**"overFlow"**: "0"**,**

**"type"**: "11"**,**

**"pageNum"**: "3"**,**

**"fontFamily"**: "SimSun"**,**

**"dateTimeFormat"**: "yyyy-MM-dd HH:mm:ss"**,**

**"name"**: "text1"**,**

**"tagType"**: "0"**,**

**"width"**: "0.2509457755359395"**,**

**"x"**: "0.10844892812105927"**,**

**"y"**: "0.07034728406055209"**,**

**"fontSize"**: "18.7"**,**

**"tag"**: "文本"**,**

**"text"**: ""**,**

**"maxLength"**: "20"**,**

**"height"**: "0.03561887800534283"**,**

**"warp"**: "true"

**},**

**{**

**"isRequired"**: "1"**,**

**"overFlow"**: "0"**,**

**"type"**: "11"**,**

**"pageNum"**: "3"**,**

**"fontFamily"**: "SimSun"**,**

**"dateTimeFormat"**: "yyyy-MM-dd HH:mm:ss"**,**

**"name"**: "text2"**,**

**"tagType"**: "0"**,**

**"width"**: "0.2509457755359395"**,**

**"x"**: "0.6620428751576293"**,**

**"y"**: "0.21015138023152272"**,**

**"fontSize"**: "18.7"**,**

**"tag"**: "文本"**,**

**"text"**: ""**,**

**"maxLength"**: "20"**,**

**"height"**: "0.05966162065894924"**,**

**"warp"**: "true"

**}**

**],**

**"pageCount"**: 8**,**

**"fileName"**: "xxxxxxxxx1201.pdf"**,**

**"developerId"**: "1852688952381669926"**,**

**"description"**: "未描述"**,**

**"titleHash"**: "fb1ae6b457f910a813fff6d0d1a6431a5b02ef17"**,**

**"title"**: "混合云—dzx—test"**,**

**"isDel"**: "0"**,**

**"tid"**: "198472374283748"

**}**

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost"**: 53**,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 4.21.5、用模版变量的手动签

说明：获取合同手动签署的页面URL，将此URL发送给指定用户，即可在此页面进行手动签操作完成签署。本接口根据模版变量定位签署位置。

Request Path：[**server**-host]/contract/sendByTemplate/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **默认值** |
| **contractId** | 合同编号 | Y |  |  |
| **signer** | 指定给哪个用户看 | Y |  |  |
| **dpi** | 预览图片清晰度，枚举值：96-低清（默认），120-普清，160-高清，240-超清 | N | 3 | 96 |
| **expireTime** | 签署链接的过期时间，unix时间戳。超过此时间则无法打开签署链接，需要获取新的签署链接。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”。  注：并不是合同的可签署到期时间，只是此签署链接的有效期。  由于返回的是短链接，如果没有设置则默认是30分钟，如果设置了以设置的为准，但有效期最大不能超过7天。 | N | 20 | 0 |
| **tid** | 模版ID | Y |  |  |
| **varNames** | 模板的变量名称，针对“模版”中的签署字段的变量名称，可以填多个。    变量名称获取方法有多种：  ①在开放平台编辑完模版后，将设置的变量名称记录下来；  ②用“4.21.0、获取模版变量”接口来获取，挑选出response中本次手动签需要用到的“type”为“50、40”的“name”字段值即需要使用的变量名称，拼接成字符串格式，不同变量名称之间用英文逗号间隔。日期类型的变量为“date”、“date1”、“date2”等date带流水号的格式。  使用示例如下：  **"varNames"**: "甲方盖章1,甲方盖章2,date,date1" | Y |  |  |
| **isAllowChangeSignaturePosition** | 在有指定signaturePOSTion参数的情况下，是否允许拖动签名位置。取值1/0。（0：不允许，1：允许） | N | 1 | 1 |
| **returnUrl** | 手动签署时，当用户签署完成后，签署页面指定回跳的页面地址，如果没填此参数则签署完成后跳转到合同预览页面。  如果returnurl带有“http”或“https”开头则判定h5跳转，如果returnurl不带“http”或“https”开头则判定为微信小程序跳转。 | N |  |  |
| **vcodeMobile** | 手动签署时指定接收手机验证码的手机号，如果需要采用手动签署页面校验短信验证码，则此项必填。  如果手动签不需要短信校验，则此项可不填。  自动签可不填此项。 | N |  | 20 |
| **isDrawSignatureImage** | 手动签署时是否手绘签名。取值0/1/2。  0点击签名图片不会触发手写面板（禁止手写）。  1点击签名图片能触发手写面板（既可手写也可使用默认签名）。  2强制必须手绘签名（只能手写不允许使用默认签名）。 | N | 1 | 1 |
| **signatureImageName** | 签名/印章图片。取值default或者指定的印章名称。（手动签都用默认的） | N | default | 50 |
| **pushUrl** | 此处有配置则签署推送消息优先使用此配置，如果此处没有配置，则取开发者统一配置的异步推送地址 | N |  |  |
| **version** | 手动签的版本，默认为2，枚举值：  2-v2版本,  3-v3版本，UI优化，移动端手绘面板自动横屏 | N | 1 | 2 |
| **isShowHandwrittenTime** | 枚举值0/1，默认为0  1-手写面板及签名图片上显示手写签名操作的时间  0-不显示该时间 | N | 1 | 0 |
| **isFaceAuth** | 是否使用刷脸签。  1-刷脸签，使用刷脸校验；  0或其他-非刷脸签，使用短信校验或无校验； | N |  |  |

**示例：**

**{**

**"contractId"**: "cU34344234"**,**

**"expireTime"**: "0"**,**

**"dpi"**: "120"**,**

**"signer"**: ""**,**

**"tid"**: "1271623948"**,**

**"varNames"**: "甲方盖章1,甲方盖章2,date,date1"**,**

**"isAllowChangeSignaturePosition"**: "0"**,**

**"returnUrl"**: ""**,**

**"vcodeMobile"**: ""**,**

**"isDrawSignatureImage"**: "1"**,**

**"signatureImageName"**: "default"**,**

**"pushUrl"**: "https://xxxxxxx"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

手动签署页面的url，签署者在此页面上进行签署操作。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"shortUrl"**: "https://..........."**,**

**"longUrl"**: "https://..........."

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 4.21.6、获取创建模版的地址

说明：使用此接口获取创建模版的页面地址，用于创建开发者模版，开发者可以向自己平台上的普通用户开放此页面。

Request Path：**[server-host]**/page/template/create/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **account** | 操作者的用户标识 | Y | 表示开发者是把这个页面提供给哪个用户操作的，这个用户标识必须已经调用“注册用户”接口在上上签系统里创建过，否则这里会报“user not exists” |

**示例：**

**{**

**"account"**: "111222333"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **url** | 创建模版的页面地址，开发者可以将此地址在自己平台上放开给用户使用 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"url"**: "https://xxxxxx"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 4.21.7、获取编辑模版的地址

说明：使用此接口获取编辑模版的页面地址，用于编辑开发者模版，开发者可以向自己平台上的普通用户开放此页面。

Request Path：**[server-host]**/page/template/modify/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **tid** | 模版编号 | Y | 从上上签开放平台的模版管理列表获取，或者通过“4.21.8、获取开发者模版列表”接口获取 |
| **account** | 操作者的用户标识 | Y | 表示开发者是把这个页面提供给哪个用户操作的，这个用户标识必须已经调用“注册用户”接口在上上签系统里创建过，否则这里会报“user not exists” |

**示例：**

**{**

**"tid"**: "111222333"**,**

**"account"**: "mlimd@163.com"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **url** | 编辑模版的页面地址，开发者可以将此地址在自己平台上放开给用户使用 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"url"**: "https://xxxxxx"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 4.21.8、获取开发者模版列表

说明：使用此接口获取编辑模版的页面地址，用于编辑开发者模版。

Request Path：**[server-host]**/template/getTemplates/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **categoryName** | 类别名称 | Y |  |
| **pageSize** | 每页显示的条数 | Y |  |
| **pageNum** | 页数 | N |  |

**示例：**

**{**

**"categoryName"**: "劳动合同"**,**

**"pageSize"**: "20"**,**

**"pageNum"**: "1"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **templates** | 模版列表字段, jsonArray格式，主要包含以下字段：  **"fid"**: "文件编号，无需关注"**,**  **"developerId"**: "开发者编号，每个模版都从属于一个开发者，每个开发者仅能查询自己的模版，一般无需关注"**,**  **"titleHash"**: "title算出来的hash，无需关注"**,**  **"title"**: "模板名称"**,**  **"isDel"**: "删除标识，0是正常，已删除的模版会删除记录，无需关注"**,**  **"tid"**: "模版编号"**,**  **"categoryName"**: "模板分类" |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"templates"**: **[**

**{**

**"fid"**: "7066638189671621264"**,**

**"developerId"**: "112"**,**

**"titleHash"**: "07f485445a900b2cb13edd961fa283e69867ebeb"**,**

**"title"**: "demo-title"**,**

**"isDel"**: "0"**,**

**"tid"**: "151264724401000001"**,**

**"categoryName"**: ""

**}**

**]**

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 4.21.9、预览模版

说明：使用此接口获取预览模版的页面地址，用于预览开发者模版的效果，开发者可以向自己平台上的普通用户开放此页面。

Request Path：**[server-host]**/page/template/preview/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **tid** | 模版编号 | Y | 从上上签开放平台的模版管理列表获取，或者通过“4.21.8、获取开发者模版列表”接口获取 |
| **account** | 操作者的用户标识 | Y | 表示开发者是把这个页面提供给哪个用户预览的，这个用户标识必须已经调用“注册用户”接口在上上签系统里创建过，否则这里会报“user not exists” |
| **sid** | 业务流水号 | N | 开发者自定义的预览流水号（比如第几次预览之类） |

**示例：**

**{**

**"tid"**: "111222333"**,**

**"account"**: "mlimd@163.com"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **url** | 预览模版的页面地址，开发者可以将此地址在自己平台上放开给用户使用 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"url"**: "https://xxxxxx"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.22、生成合同附页

说明：当合同结束后，可以调用此接口为合同生成一个附页文件，该附页文件是上上签提供的对合同从创建到签署完毕的整个签署过程的证明。

Request Path：**[server-host]**/contract/createAttachment

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **contractId** | 合同编号 | Y |  |

**示例：**

**{**

"contractId"**:** "1703465339807805447"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.23、下载合同附页文件

说明：用此接口可以下载附页文件。

Request Path：**[server-host]**/contract/downloadAttachment

Method：GET

其他url参数：**contractId**

需要参与签名的其他URL参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **contractId** | 合同编号 | Y |  |

**请求参数（REQUEST BODY）：无**

**示例：**

**GET [**server-host**]/**contract**/**downloadAttachment**/?**developerId**=**d445f5432**&**rtick**=**17364899234**&**sign**=**a5334df21323**&**signType**=**rsa**&**contractId**=**d445f5432

**响应（RESPONSE）：**

正常示例：

**返回文件流**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.24、关键字定位签署合同

说明：签署合同，可直接用于无感知的快捷签署。此接口要求签署者必须拥有自己的数字证书。可以无需先调用“添加签署者”接口与“设置合同签署参数”接口。

Request Path：**[server-host]**/contract/sign/keywords/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **contractId** | 合同编号 | Y |  |  |
| **signerAccount** | 签署者 | Y |  | 即签署者的account |
| **keywords** | 关键字列表，array格式 | Y |  | 需要添加签名位置的关键字列表，支持多个关键字。内部以String数组进行拼接，示例如下： ["张三","乙方"] |
| **align** | 对齐方式 | N |  | 签名图片/印章图片与关键字的对齐方式，枚举值如下：  lt 左上角对齐（图片位于关键字左上角）  lc 左中对齐（图片位于关键字左侧上下居中）  lb 左下对齐（图片位于关键字左下角）  ct 中上对齐（图片位于关键字上方左右居中）  cc 正中对齐（图片与关键字中心对齐）  cb 中下对齐（图片位于关键字下方左右居中）  rt 右上角对齐（图片位于关键字右上角）  rc 右中对齐（图片位于关键字右侧上下居中）  rb 右下对齐（图片位于关键字右侧下角） |
| **signatureImageData** | 签名图片 | N |  | 用户指定的签名图片，Base64字符串。允许为空，为空时使用用户默认的签名图片。  使用自动签的用户，可以越过“2.1、创建用户签名/印章图片”接口、或“2.2、上传用户签名/印章图片”接口，直接在此提供签名/印章图片即可。 |
| **signatureImageName** | 签名图片名称 |  |  | 签名图片名称，取自【2.2上传用户签名/印章图片】时设置的imageName，适用于多签章场景，可在签署时灵活使用不同的签章。 |
| **signatureImageWidth** | 图片显示宽度 |  | N | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 |
| **signatureImageHeight** | 图片显示高度 |  | N | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 |

**示例：**

**{**

**"contractId"**: "150347901101000001"**,**

**"signerAccount"**: "13456833929"**,**

**"keywords"**: **[**

"张三"**,**

"乙方"

**],**

**"align"**: "cc"**,**

**"signatureImageData"**: ""**,**

**"signatureImageName"**: ""

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.25、在线验签（通过合同ID和哈希值）

说明：

提供上上签的合同编号、已签署完成的PDF文件哈希值，通过校验该哈希值与合同编号的匹配关系验证PDF文件是否有被篡改。

Request Path：**[server-host]**/contract/verifyContractFileHash

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **contractId** | 合同编号 | Y |  | 通过上上签创建的合同编号 |
| **fhash** | 文件哈希值 | Y |  | 已签署完成的合同PDF文件的哈希值 |

**示例：**

{

**"contractId"**: "18724324823748723" ,

**"fhash"**: "ZHNmYXNnZHNnZHNhZ3Nn.."

}

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |  |
| --- | --- | --- |
| **参数名** | **说明** |  |
| **result** | 验证结果 | 1-文档未被篡改  0-除未被篡改之外所有的结果都为0（如一倍篡改、不是通过上上签签署完成的PDF文件等） |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"result"**: "1"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.26、获取存证页URL

说明：通过此接口可以获取到存证页面的url，此页面表示该合同已在上上签保存。

* 此页面可联系上上签进行配置，添加跳转链接（比如仲裁网站），默认跳转到上上签官网。
* 合同尚未结束时（即还有签署者未签署完成，没有调用过结束合同接口），无法获取此页面url。
* 如果签署者未申请过数字证书，则页面上关于签署者的证件信息会显示不详。

Request Path：[**server-host**]/contract/getApplyArbitrationURL/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **默认值** |
| **contractId** | 合同编号 | Y |  |  |
| **account** | 指定给哪个用户看 | Y |  |  |

**示例：**

**{**

"contractId"**:** "cU34344234"**,**

"account"**:** ""

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

合同存证页面的url，签署者在此页面上查看合同编号、签署者基本信息等。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"url"**: "https://..........."

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

## 12、支付服务

### 12.1、微信支付

说明：通过此接口可以获取微信支付的二维码，通过展示二维码进行支付。

Request Path：[**server-host**]/pay/wechatPay/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **默认值** |
| **contractId** | 合同编号，表示这笔支付订单用于支付哪一份合同 | N |  |  |

**示例：**

**{**

**"contractId"**: "cU34344234"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **qrCodeImageData** | 微信支付使用的二维码图片，Base64编码，需解码后展示。有30分钟的有效期，如果超过30分钟未支付则需重新获取二维码图片 |
| **orderId** | 支付订单编号 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"qrCodeImageData"**: "xxxxxx",

**"orderId"**: "xxxxxx",

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 12.1、查询支付订单

说明：通过支付订单编号查询支付结果。

Request Path：[**server-host**]/pay/getOnlinePayOrder/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **默认值** |
| **orderId** | 支付订单编号 | Y |  |  |

**示例：**

**{**

**"orderId"**: "2130366160674232326"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |  |
| --- | --- | --- |
| **参数名** | **说明** |  |
| **orderId** | **订单编号** |  |
| **paymentMode** | **该订单所使用的支付方式** | WECHATPAY——微信支付  ALIPAY——支付宝  CHINAPAY——银联电子 |
| **completeDateTime** | **订单完成时间** | **格式：yyyy-MM-dd HH:mm:ss** |
| **orderAmount** | **订单支付的金额（元）** |  |
| **payStatus** | **支付状态，只有状态为“SUCCESS”时才是支付成功** | **如果是微信支付，状态为：**SUCCESS  FAIL  UNPAID  PROCESSING  UNDO  SYSTEM\_BUSY  UNDEFINED |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"orderId"**: "2130366160674232326"**,**

**"paymentMode"**: "WECHATPAY"**,**

**"completeDateTime"**: "2018-11-21 18:00:53"**,**

**"orderAmount"**: "0.4"**,**

**"payStatus"**: "SUCCESS"

**},**

**"cost"**: 140

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

# 五、选配API功能列表

## 1、用户服务

### 1.2、注册企业用户并申请证书（支持CFCA企业证书）

说明：创建一个上上签企业用户，并提交用户的证件信息，为该用户申请CFCA的企业数字证书。后续有关该用户的所有操作都使用该用户的account。

由于证书是异步申请，不会立即返回证书申请结果，所以在签署之前，需要确认一次该用户的证书是否申请成功，调用接口“1.11、异步申请状态查询”确认。

Request Path：[**server-host**]/user/reg/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | | | |
| **account** | 用户唯一标识 | Y |  | 用户的唯一标识，可以是邮箱、手机号、证件号等不限，建议使用邮箱或手机号码。 | | | | |
| **name** | 企业名称 | Y | 100 | 必须和企业证件上登记的名称一致  如个体工商户在营业执照上无企业名称的，请填“经营者”名字 | | | | |
| **userType** | 用户类型 | Y | 1 | 2表示企业 | | | | |
| **mail** | 用户邮箱 | N | 50 |  | | | | |
| **mobile** | 用户手机号 | N | 20 |  | | | | |
| **credential** | 企业证件信息对象 | Y |  | 企业证件信息对象，json格式  三证合一的统一社会信用代码这样传：regcode = "统一社会信用代码"orgcode = "统一社会信用代码"taxcode = "统一社会信用代码"老三证这样传：regcode = "工商注册号"orgcode = "组织机构代码证"taxcode = "税务登记证"个体户这样传：regcode = "工商注册号"orgcode = ""taxcode = ""  具体参数如下： | | | | |
| **参数** | **说明** | **是否必填** | **最大长度** |  |
| **regCode** | 工商注册号 | Y | 30 | 企业用户设置 |
| **orgCode** | 组织机构代码 | Y | 30 | 企业用户设置  regCode和orgCode必须输入一个，否则异常返回：240006: regCode/orgCode must be have one |
| **taxCode** | 税务登记证号 | Y | 30 | 企业用户设置 |
| **legalPerson** | 法定代表人姓名 | Y | 50 | 法定代表人姓名或经办人姓名 |
| **legalPersonIdentity** | 法定代表人证件号 | Y | 30 | 法定代表人证件号或经办人证件号 |
| **legalPersonIdentityType** | 法定代表人证件类型 | Y | 2 | 法定代表人证件类型或经办人证件类型，与“**legalPersonIdentity**”要匹配，默认为“0”。  0-居民身份证  1-护照  B-港澳居民往来内地通行证  C-台湾居民来往大陆通行证  E-户口簿  F-临时居民身份证 |
| **legalPersonMobile** | 法定代表人手机号 | N | 20 | 法定代表人手机号或经办人手机号 |
| **contactMobile** | 联系手机 | Y | 20 | 联系手机必填，为CA年检抽查时联系使用 |
| **contactMail** | 联系邮箱 | N | 50 | **credential对象属性** |
| **province** | 省份 | N | 10 | **credential对象属性** |
| **city** | 城市 | N | 50 | **credential对象属性** |
| **address** | 地址 | N | 100 | **credential对象属性** |
| **applyCert** | 是否申请证书 | Y | 1 | 申请填写为1 | | | | |
| **certType** | 证书类型 | N |  | 填“CFCA”即可申请CFCA的企业证书，申请CFCA企业证书会产生额外的费用，使用前请先联系上上签商务确定费用事宜，再通知上上签项目经理开通权限。 | | | | |

**示例：**

**{**

**"account"**: "test@bestsign.cn"**,**

**"mail"**: "test@bestsign.cn"**,**

**"mobile"**: "13800001234"**,**

**"name"**: "浙江xx公司"**,**

**"userType"**: "2"**,**

**"credential"**: **{**

**"regCode"**: "5566789"**,**

**"taxCode"**: "5566789"**,**

**"orgCode"**: "5566789"**,**

**"contactMail"**: "test@bestsign.cn"**,**

**"contactMobile"**: "13800001234"**,**

**"legalPerson"**: "张三"**,**

**"legalPersonIdentity"**: "100123199001010011"**,**

**"legalPersonIdentityType"**: "0"**,**

**"legalPersonMobile"**: "17712341234"**,**

**"province"**: "浙江省"**,**

**"city"**: "杭州市"**,**

**"address"**: "xx路xx号"

**},**

**"applyCert"**: "1"

**}**

**响应（RESPONSE）：**

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **taskId** | 异步申请证书队列中的任务编号，在24小时内可用于查询异步申请状态，**taskId**过24小时后就失效 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"taskId"**: "170217170217942"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 1.6、注册用户

说明：创建一个上上签用户，后续有关该用户的所有操作都使用该用户的account。

Request Path：[**server-host**]/user/reg/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 用户的唯一标识，可以是邮箱、手机号、证件号等不限，建议使用邮箱或手机号码。 |
| **name** | 用户名称 | Y | 100 | 如姓名太长则生成系统字体签名时会显示不全，图片上最多显示20字 |
| **userType** | 用户类型 | Y | 1 | 1表示个人，2表示企业 |
| **mail** | 邮箱 | N | 50 |  |
| **mobile** | 手机号 | N | 20 |  |

**示例：**

**{**

**"account"**: "test@bestsign.cn"**,**

**"mail"**: "test@bestsign.cn"**,**

**"mobile"**: "13800001234"**,**

**"name"**: "杭州上上签"**,**

**"userType"**: "2"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"data"**: **{ },**

**"cost": 233,**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 1.7、设置个人用户证件信息

说明：为某一个个人用户设置证件信息，该证件信息用于申请数字证书。通常用在“1.1、注册个人用户并申请证书”由于证件信息错误导致的容错重试。

Request Path：[**server-host**]/user/setPersonalCredential/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 为哪个用户设置证件信息就填该用户的唯一标识 |
| **identity** | 证件号 | Y | 30 | 需要和证件上登记的号码一致 |
| **name** | 姓名 | Y | 50 | 需要和证件上登记的姓名一致 |
| **identityType** | 证件类型 | N | 2 | 默认为“0”, "0"表示身份证 |
| **contactMobile** | 联系手机 | N | 20 |  |
| **contactMail** | 联系邮箱 | N | 50 |  |
| **province** | 省份 | N | 32 |  |
| **city** | 城市 | N | 50 |  |
| **address** | 地址 | N | 100 |  |

**示例：**

**{**

**"account"**: "u-1343423222"**,**

**"identity"**: "100123199001010011"**,**

**"identityType"**: "0"**,**

**"contactMail"**: "test@bestsign.cn"**,**

**"contactMobile"**: "13800001234"**,**

**"name"**: "张三"**,**

**"province"**: "浙江省"**,**

**"city"**: "杭州市"**,**

**"address"**: "xx路xx号"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"data"**: **{ },**

**"cost": 233,**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 241208**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "user exists, can't add again."

**}**

### 1.8、设置企业用户证件信息

说明：为某一个企业用户设置证件信息，该证件信息用于申请数字证书。通常用在“1.2、注册企业用户并申请证书”由于证件信息错误导致的容错重试。

Request Path：[**server-host**]/user/setEnterpriseCredential/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 为哪个用户设置证件信息就填该用户的唯一标识 |
| **regCode** | 工商注册号 | Y | 30 | 企业用户设置  三证合一的统一社会信用代码这样传：  regcode = "统一社会信用代码"  orgcode = "统一社会信用代码"  taxcode = "统一社会信用代码"  老三证这样传：  regcode = "工商注册号"  orgcode = "组织机构代码证"  taxcode = "税务登记证"  个体户这样传：  regcode = "工商注册号"  orgcode = ""  taxcode = "" |
| **orgCode** | 组织机构代码 | Y | 30 | 企业用户设置  regCode和orgCode必须输入一个，否则异常返回：240006: regCode/orgCode must be have one |
| **taxCode** | 税务登记证号 | Y | 30 | 企业用户设置 |
| **name** | 企业名称 | Y | 50 | 需要和证件上登记的名称一致  如个体工商户无企业名称的，则请填写证件上的“经营者”名字 |
| **legalPerson** | 法定代表人姓名 | Y | 50 | 法定代表人姓名或经办人姓名 |
| **legalPersonIdentity** | 法定代表人证件号 | Y | 30 | 法定代表人证件号或经办人证件号 |
| **legalPersonIdentityType** | 法定代表人证件类型 | Y | 2 | 法定代表人证件类型或经办人证件类型，与“**legalPersonIdentity**”要匹配，默认为“0”。  0-居民身份证  1-护照  B-港澳居民往来内地通行证  C-台湾居民来往大陆通行证  E-户口簿  F-临时居民身份证 |
| **legalPersonMobile** | 法定代表人手机号 | N | 20 | 法定代表人手机号或经办人手机号 |
| **contactMobile** | 联系手机 | Y | 20 | 联系手机必填，为CA年检抽查时联系使用 |
| **contactMail** | 联系邮箱 | N | 50 |  |
| **province** | 省份 | N | 32 |  |
| **city** | 城市 | N | 50 |  |
| **address** | 地址 | N | 100 |  |

**示例：**

**{**

**"account"**: "u-1343423222"**,**

**"regCode"**: "5566789"**,**

**"taxCode"**: "5566789"**,**

**"orgCode"**: "5566789"**,**

**"contactMail"**: "test@bestsign.cn"**,**

**"contactMobile"**: "13800001234"**,**

**"name"**: "浙江xx公司"**,**

**"legalPerson"**: "张三"**,**

**"legalPersonIdentity"**: "100123199001010011"**,**

**"legalPersonIdentityType"**: "0"**,**

**"legalPersonMobile"**: "17712341234"**,**

**"province"**: "浙江省"**,**

**"city"**: "杭州市"**,**

**"address"**: "xx路xx号"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"data"**: **{ },**

**"cost": 233,**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240006**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "legalPersonIdentity wrong"

**}**

### 1.9、同步申请数字证书

说明：

1. 为用户申请数字证书，该数字证书用于电子签名。
2. 同步申请数字证书前必须先设置用户证件信息（调用“1.7、设置个人用户证件信息”或“1.8、设置企业用户证件信息”接口），通常用于“1.1”或者“1.2”的容错重试。
3. 用户必须先申请数字证书，否则将无法完成签署。

Request Path：[**server-host**]/user/applyCert/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 为哪个用户申请证书就填该用户的唯一标识 |

**示例：**

**{**

**"account"**: "mlimd@163.com"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **certId** | 证书编号 |
| **certType** | 证书类型 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"certType"**: "CFCA"**,**

**"certId"**: "CFCA-33-20170217170217942-65183"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 1.13、申请专用数字证书

说明：

1. 为用户唯一标识申请专用的数字证书，该数字证书用于电子签名，并且在上上签的体系内是该用户独有的，不会被其他开发者用相同证件信息申请证书的行为复用到该证书。
2. 申请专用数字证书前必须先调用“1.6、注册用户”，然后设置用户证件信息（调用“1.7、设置个人用户证件信息”或“1.8、设置企业用户证件信息”接口）。

Request Path：[**server-host**]/user/applyCert/unique/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 为哪个用户申请证书就填该用户的唯一标识 |

**示例：**

**{**

**"account"**: "mlimd@163.com"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **certId** | 证书编号 |
| **certType** | 证书类型 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"certType"**: "CFCA"**,**

**"certId"**: "CFCA-33-20170217170217942-65183"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 1.14、吊销专用数字证书

说明：删除专用证书的私钥，并吊销该专用证书。

Request Path：[**server-host**]/user/revokeCert/unique/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 为哪个用户申请证书就填该用户的唯一标识 |

**示例：**

**{**

**"account"**: "mlimd@163.com"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **result** | "1"表示成功， "0"表示失败 |
| **certType** | error |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"result"**: "0"**,**

**"error"**: "吊销失败"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

## 3、文件存储服务

### 3.2、转换文件格式

说明：

文件格式转换，只支持从doc, docx转换到PDF，需要转换的文件必须先通过接口【3.1、上传合同文件】得到fid。

Request Path：**[server-host]**/storage/convert/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y | 64 | 必须要指定一个用户唯一标识作为操作者 |
| **fid** | 源文件编号 | Y | 20 | 通过“3.1、上传合同文件”得到 |
| **ftype** | 文件类型 | N | 10 | 要转换成的目标文件类型，目前只支持PDF。默认值PDF |

**示例：**

**{**

**"account"**: "13456833929"**,**

**"fid"**: "1809789296522035810"**,**

**"ftype"**: "PDF"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **fid** | 文件编号 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"fid"**: "f1343423222"

**}**

**}**

异常示例：

**{**

**"errno"**: 240006**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "only support doc/docx/xls/xlsx/ppt/pptx"

**}**

## 4、单文档合同服务

### 4.2、签署合同（即自动签，每个位置可用不同的图片）

说明：签署合同，可直接用于无感知的快捷签署。此接口要求签署者必须拥有自己的数字证书。可以无需先调用“添加签署者”接口与“设置合同签署参数”接口。

Request Path：**[server-host]**/contract/sign/cert/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | |
| **contractId** | 合同编号 | Y |  |  | | |
| **signer** | 签署者 | Y |  | 即签署者的account | | |
| **signaturePositions** | 指定的签署位置，json array格式 | Y |  | 每一项为一个json对象 | | |
|  |  | 最大长度 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| y | 纵坐标，同上 | 10 |
| pageNum | 页码。1开始，不能超过实际页数 | 3 |
| rptPageNums | 当前位置的签名需要复制到的目标页码列表。该参数用于控制是否将当前位置的签名复制到其他页。参数启用之后，当前页的签名会同时出现在其他页的相同位置。参数值格式如下： 多个页码使用英文逗号(,)分隔。 参数不存在或者参数值为空表示不复制， 参数值为0表示复制到所有页， 参数值为其他表示复制到指定页。 参数值示例："0" "1,2,3,4,5,6" "1,3,5,7,9" |  |
| signatureImageData | 用户指定的签名图片，Base64字符串。表示允许每个位置使用不同的图片。允许为空，为空或者图片数据错误时使用外层的图片设置。 |  |
| signatureImageWidth | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。  目前如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。宽高为配套参数，非必填参数，此处不设置则使用外层的宽高设置。 |  |
| signatureImageHeight | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。  目前如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。宽高为配套参数，非必填参数，此处不设置则使用外层的宽高设置。 |  |
| **signatureImageName** | 签名图片名称 | N |  | 企业用户可以指定上传过的印章图片中的某一张作为本次签署的印章图片，取“上传用户签名/印章图片”接口设置的**imageName**的值 | | |
| **signatureImageData** | 签名图片 | N |  | 用户指定的签名图片，Base64字符串。允许为空，为空时使用用户默认的签名图片。  使用自动签的用户，可以越过“2.1、创建用户签名/印章图片”接口、或“2.2、上传用户签名/印章图片”接口，直接在此提供签名/印章图片即可。 | | |
| **signatureImageWidth** | 签名图片显示宽度 | N |  | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。 同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。 如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | | |
| **signatureImageHeight** | 签名图片显示高度 | N |  | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | | |

**示例：**

**{**

**"contractId"**: "cU34344234"**,**

**"signer"**: "u-1112343242"**,**

**"signatureImageData"**: "cU34344234"**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.05"**,**

**"y"**: "0.71"**,**

**"rptPageNums"**: "2,3,4,5,6,7"**,**

**"signatureImageData"**: "图片base64"**,**

**"signatureImageWidth"**: "100"**,**

**"signatureImageHeight"**: "70"

**},**

**{**

**"pageNum"**: "2"**,**

**"x"**: "0.05"**,**

**"y"**: "0.81"**,**

**"signatureImageData"**: "图片base64"**,**

**"signatureImageWidth"**: "100"**,**

**"signatureImageHeight"**: "70"

**}**

**]**

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.12、单个添加签署者

说明：为已创建的合同添加签署者，添加签署者之后的合同正式发送。

Request Path：[**server-host**]/contract/addSigner/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **contractId** | 合同编号 | Y | 创建合同 |
| **signer** | 要添加的签署者account | Y |  |

**示例：**

**{**

**"contractId"**: "17026986666669878"**,**

**"signer"**: "abc@163.com"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"data"**: **{ },**

**"cost": 233,**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 241207**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "user not exist"

**}**

### 4.13、批量添加签署者

说明：为已创建的合同批量添加签署者，添加签署者之后的合同正式发送。

Request Path：[**server-host**]/contract/addSigners/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **contractId** | 合同编号 | Y |  |
| **signers** | 要添加的签署者account | Y | json array格式 |

**示例：**

**{**

**"contractId"**: "1702698069495119878"**,**

**"signers"**: **[**

"13867410069"**,**

"13067825582"

**]**

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

返回各account添加成功与否情况。格式为"account":"1/0"，成功为"1"，失败为"0"。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"13067825582"**: "1"**,**

**"13867410069"**: "1"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 4.14、设置合同签署参数

说明：为签署者设置合同签署参数，包括签署位置的坐标（如坐标为设置则以签署接口为准）、使用的签名/印章图片等。

Request Path：[**server-host**]/contract/setSignerConfig/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | | | **是否必填** | **默认值** | **最大长度** |
| **contractId** | 合同编号 | | | Y |  |  |
| **account** | 签署者 | | | Y |  |  |
| **signaturePositions** | 指定的默认签署位置，json array格式，每一项为一个json对象。如用手动签模式则此项必填。  x,y坐标使用百分比，取值0.0 - 1.0。如果不指定此参数，表示由签署者自行在页面上选一个签署位置 | | | N |  |  |
| **参数名** | **说明** | **最大长度** |
| y | 纵坐标，同上 | 10 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| pageNum | 页码，从1开始，不能超过合同最大页数 | 3 |
| **isAllowChangeSignaturePosition** | 在有指定signaturePOSTion参数的情况下，是否允许拖动签名位置。取值1/0。（0：不允许，1：允许） | | | N | 1 | 0 |
| **returnUrl** | 手动签署时，当用户签署完成后，签署页面指定回跳的页面地址，如果没填此参数则签署完成后跳转到合同预览页面。  如果returnurl带有“http”或“https”开头则判定h5跳转，如果returnurl不带“http”或“https”开头则判定为微信小程序跳转。 | | | N |  |  |
| **password** | 合同密码（暂不支持） | | | N |  | 18 |
| **isVerifySigner** | 手动签署时是否验证account身份。取值0/1。如果要验证account身份，则忽略"查看密码"和"验证码"的设置（暂不支持） | | | N | 0 | 1 |
| **vcodeMail** | 手动签署时指定接收邮件验证码的邮箱。（暂不支持） | | | N |  | 50 |
| **vcodeMobile** | 手动签署用来接收短信验证码的手机号。  如手动签需要校验短信验证码则必填此项。  如不填写则手动签时不校验短信验证码。 | | | N |  | 20 |
| **isDrawSignatureImage** | 手动签署时是否手绘签名。取值0/1/2。  0点击签名图片不会触发手写面板（禁止手写）。  1点击签名图片能触发手写面板（既可手写也可使用默认签名）。  2强制必须手绘签名（只能手写不允许使用默认签名）。 | | | N | 1 | 1 |
| **signatureImageName** | 签名/印章图片。取值default或者指定的印章名称。（手动签都用默认的） | | | N | default | 50 |
| **app** | 与客户app交互所需的JSBridge（暂不支持） | | | N |  |  |
| **isShowHandwrittenTime** | 枚举值0/1，默认为0  1-手写面板及签名图片上显示手写签名操作的时间  0-不显示该时间 | | | N | 1 | 0 |
| **isFaceAuth** | 是否使用刷脸签。  1-刷脸签，使用刷脸校验；  0或其他-非刷脸签，使用短信校验或无校验； | | | N |  |  |

**示例：**

**{**

"contractId"**:** "cU34344234"**,**

"account"**:** "u-34234234324"**,**

"signaturePositions"**:** **[**

**{**

"pageNum"**:** "1"**,**

"x"**:** "0.05"**,**

"y"**:** "0.01"

**},**

**{**

"pageNum"**:** "2"**,**

"x"**:** "0.05"**,**

"y"**:** "0.01"

**}**

**],**

"isAllowChangeSignaturePosition"**:** "0"**,**

"returnUrl"**:** ""**,**

"password"**:** ""**,**

"isVerifySigner"**:** "0"**,**

"vcodeMail"**:** ""**,**

"vcodeMobile"**:** ""**,**

"isDrawSignatureImage"**:** "1"**,**

"signatureImageName"**:** "default"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.15、结束合同

说明：

* 当所有签署者都签署完毕后，使用此接口结束整个合同签署过程。当一份合同所有签署者都签署完成后，必须调用结束合同，否则合同状态一直为“等待签署中”。只有结束的合同才能提供公证服务。
* 结束合同不会打上上签的系统签名。
* 此接口常用于这样的场景：一份合同，甲方用上上签来签署，乙方用其他电子签约系统来签署，这时候甲方签署完的合同不能进行锁定，需要用此接口来结束合同，表示在上上签系统内的签署过程已经完毕。
* “锁定结束合同”与“结束合同”为二选一使用的接口。

Request Path：**[server-host]**/storage/contract/end/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **contractId** | 合同编号 | Y |  |

**示例：**

**{**

"contractId"**:** "1703465339807805447"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.16、骑缝章签署

说明：使用骑缝章签署合同。

Request Path：**[server-host]**/storage/contract/cert/paging/seal/sign/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **signer** | 帐号 | Y | 64 | 使用骑缝章的签署者 |
| **signY** | 骑缝章Y坐标 | Y | 100 | 骑缝章在页面的y坐标 |
| **contractId** | 合同编号 | Y | 20 |  |
| **signatureImageData** | 骑缝章图片 | Y | 2048000 | 签名图片Base64字符串。 |

**示例：**

**{**

"contractId"**:** "150208027001000001"**,**

"signer"**:** "wanghf617@qq.com"**,**

"signatureImageData"**:** ""**,**

"signY"**:** "0.45"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **fid** | 新文件ID |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"fid"**: "1694593128888888"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 4.17、删除合同文件

说明：

* 当合同签署结束，开发者通过此接口将属于本开发者的合同文件彻底删除，仅保留签署记录。
* 因删除合同风险太大，“用钱宝”的商务合同中约定了删除的条款，此接口仅供“用钱宝”使用。

Request Path：**[server-host]**/storage/contract/delete/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **contractId** | 合同编号 | Y |  |

**示例：**

**{**

**"contractId"**: "1703465339807805447"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.18、获取合同手动签署页URL

说明：获取合同手动签署的页面URL，可以根据设置的合同签署参数选择进行短信校验、手绘签名等操作，并在此页面完成签署。

Request Path：[**server-host**]/contract/getSignURL/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **默认值** |
| **contractId** | 合同编号 | Y |  |  |
| **account** | 指定给哪个用户看 | Y |  |  |
| **dpi** | 预览图片清晰度，枚举值：96-低清（默认），120-普清，160-高清，240-超清 | N | 3 | 96 |
| **expireTime** | 签署链接的过期时间，unix时间戳格式（秒数）。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”。超过此时间则无法打开链接页面，需要获取新的签署合同链接。  注：并不是合同的可签署到期时间，只是此签署链接的有效期。最长不能超过当前时间的7天。 | N | 20 | 0 |
| **sid** | 开发者自定义的签署流水号（比如第几次签署之类），如有填写，则在异步推送签署结果时包含此签署流水号 | N |  |  |
| **version** | 手动签的版本，默认为2，枚举值：  2-v2版本,  3-v3版本，UI优化，移动端手绘面板自动横屏 | N | 1 | 2 |
| **readAll** | 枚举值0/1，默认为0  0-无需拖动到页面底部即可确认签署  1-必须拖动到页面底部，表示阅读完毕才能确认签署  （此处暂未生效） | N |  |  |

**示例：**

**{**

**"contractId"**: "cU34344234"**,**

**"expireTime"**: "0"**,**

**"dpi"**: "120"**,**

**"account"**: ""

**}**

**响应（RESPONSE）：**errno返回0则为正常，errno返回错误码则为异常，此时观察具体的errmsg具体分析。errno与errmsg请参阅本文《四、附录：返回信息编号列表》，返回信息编号列表持续更新。

RESPONSE DATA参数如下：

手动签署页面的url，签署者在此页面上进行签署操作。

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"data"**: **{**http://tool.oschina.net/tools/json_format/Expanded.gif

**"url"**: "https://..........."

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.19、拒绝签署

说明：当某个签署者对合同表示有问题，不想继续签署时，或者当发起人发现有问题需要取消合同时，使用此接口结束合同。拒绝签署后此合同结束，其他未签署的签署者无法继续操作。

Request Path：[**server-host**]/contract/refuse/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **contractId** | 合同编号 | Y |  |
| **signer** | 签署者的account | Y |  |
| **refuseReason** | 拒绝原因 | Y | 200 |

**示例：**

**{**

"contractId"**:** "1703466073594003458"**,**

"signer"**:** "mlimd@163.com"**,**

"refuseReason"**:** "不要吃兔兔~"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

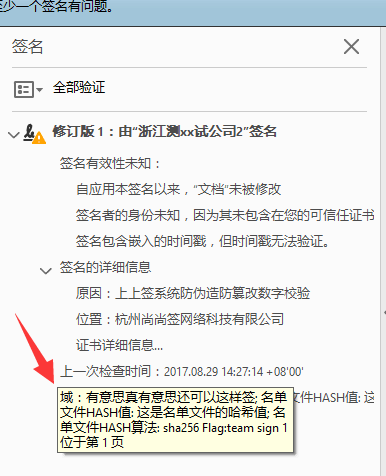
**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.20、团签

说明：

* 团签通常用于一份合同有大批签署人（人数通常在200个以上）需要签署，一旦签署人数过多，有可能会导致Adobe 的pdf阅读器打开合同pdf时，从第255个签名提示签名无效，因此采用一种新的方式进行签名。
* 使用此接口，需要准备一个文件，此文件内容为本次团签对应的所有签署人的名单，因为人数太多所以用一次团签来表示了。然后将此名单文件计算哈希值，将哈希值与哈希算法通过接口参数提交，用于后期的核对。核对方法是开发者提供名单文件，按照相同的哈希算法计算哈希值，比较哈希值是否一致，如果一致则证明名单没有变化，然后可以打开名单文件查阅这次团签对应的所有签署人是哪些。
* 由于团签是采用的名单文件代替单个签署人进行签署的方式，因此本接口不需要调用“添加签署人”、以及“设置合同签署参数”接口。
* 团签的签名域显示效果如下图所示：

Request Path：**[server-host]**/storage/contract/team/sign/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | |
| **contractId** | 合同编号 | Y |  |  | | |
| **signImage** | 签名图片文件 | Y |  | Base64 | | |
| **certId** | 证书编号 | Y |  | 团签使用的证书编号，一般固定使用开发者自己的证书，用接口“1.3、查询证书编号”获取后填入 | | |
| **nameList** | 自定义域文本 | Y |  | 团签这个签名域的自定义文本，即一个自定义的字符串，可以写诸如“XX项目第X批投资” | | |
| **nameListHash** | 哈希值 | Y |  | 名单文件的哈希值 | | |
| **hashAlgorithm** | 哈希算法 | Y |  | 名单文件哈希值的哈希算法 | | |
| **signaturePositions** | 指定的签署位置，json array格式 | Y |  | 每一项为一个json对象 | | |
|  |  | 最大长度 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| y | 纵坐标，同上 | 10 |
| pageNum | 页码。1开始 | 3 |
| **signIp** | 签署IP | N |  | 使用团签这个接口的用户的IP地址，用于表示此团签是在哪个IP进行签署的 | | |

**示例：**

**{**

**"contractId"**: "1796947239701053444"**,**

**"signImage"**: "file\_base64"**,**

**"certId"**: "CFCA-33-20170809142751732-81453"**,**

**"nameList"**: "test1;test2;test3;test4;test5;test6;test7;test8"**,**

**"nameListHash"**: "testHastvalue"**,**

**"hashAlgorithm"**: "sha256"**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.3"**,**

**"y"**: "0.5"

**}**

**],**

**"signIp"**: "127.0.0.1"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

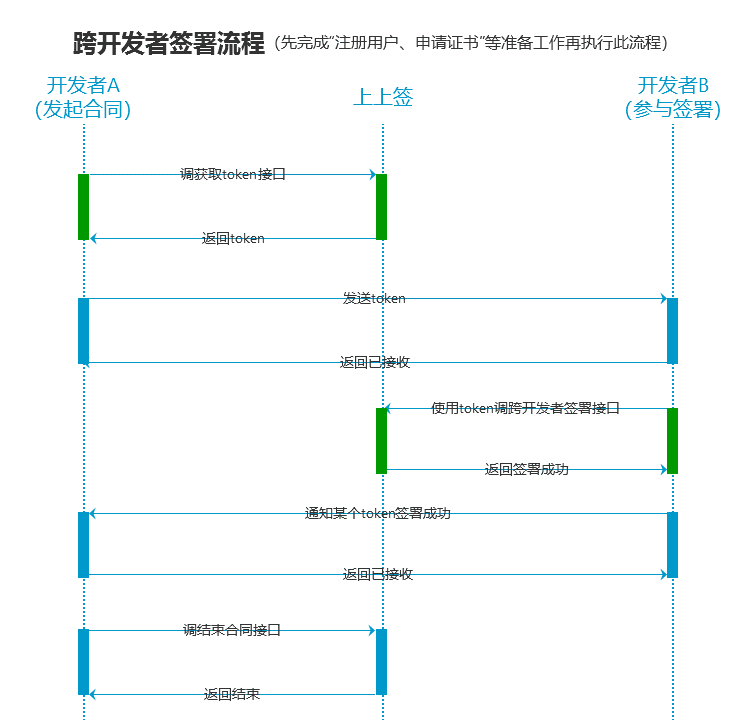
**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.27、跨开发者签署模式

跨开发者签署是一种特殊的签署模式，流程见下图所示：



#### 4.27.1、获取跨开发者签署token

说明：通过此接口可以获取跨开发者签署的token，给到其他开发者，其他开发者就可以使用此token对指定的合同进行签署。

Request Path：[**server-host**]/contract/cross/token/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **contractId** | 合同编号 | Y |  |  |

**示例：**

**{**

**"contractId"**: "1841108809792094215"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| crossToken | 跨平台签署的token，可以发给其他开发者去签署 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"crossToken"**: "P+yWSxfEVUpcUjzFDBHnv8U2e8h8KRNhR0J8NLtfOHZRUyeTlgtj2QOefaBPkLgtXhl4PoOaPudvXcSPl288TD9R92LTHTnITxxzuDvLMD3KpstDHWGFcfnc/y40XIBK"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 4.27.2、跨开发者签署

说明：开发者A（合同的发起者）发送跨开发者签署的token给开发者B（参与签署的其他开发者），开发者B就能使用此token签署A指定的合同。

Request Path：[**server-host**]/contract/cross/sign/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | |
| **signer** | 签署者 | Y |  | 即签署者的account | | |
| **crossToken** | 跨平台token | Y |  | 合同所有者生成并分发 | | |
| **signImageName** | 签章名称 | N |  | 不填写，则用默认该用户默认签章 | | |
| **signaturePositions** | 指定的签署位置，json array格式 | Y |  | 每一项为一个json对象 | | |
| 参数 | 说明 | 最大长度 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| y | 纵坐标，同上 | 10 |
| pageNum | 页码。1开始 | 3 |

**示例：**

**{**

**"signer"**: "u-1112343242"**,**

**"crossToken"**: "cU34344234"**,**

**"signImageName"**: "合同专用章"**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.05"**,**

**"y"**: "0.01"

**},**

**{**

**"pageNum"**: "2"**,**

**"x"**: "0.05"**,**

**"y"**: "0.01"

**}**

**]**

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 4.27.3、跨开发者下载

开发者A（合同的发起者）发送跨开发者操作的token给开发者B（参与签署的其他开发者），开发者B就能使用此token下载A指定的合同。

Request Path：**[server-host]**/contract/cross/download/

|  |  |
| --- | --- |
| 请求类型 | GET |
| 其他URL参数 | 无 |
| 需要参与签名的其他URL参数 | 无 |
| REQUEST BODY | 无 |
| RESPONSE DATA | 返回文件数据流 byte[] |
| 示例 | http://localhost:8080/openapi/v3/contract/cross/download?developerId=1&crossToken=HuhwQqHYv6ezY4J1v%2FMf97g2e8h8KRNhR0J8NLtfOHZRUyeTlgtj2QOefaBPkLgtXhl4PoOaPudvXcSPl288TD9R92LTHTnITxxzuDvLMD3KJ2eqM2j2nKooHBzcn68R |

### 4.28、合同批量签署（即批量自动签）

说明：签署合同，可直接用于无感知的快捷签署。适用于同一份合同内的多个签署者同时签署，同一个签署者使用同一张签名/印章图片。

此接口要求签署者必须拥有自己的数字证书。可以无需先调用“添加签署者”接口与“设置合同签署参数”接口。

Request Path：**[server-host]**/contract/sign/cert/batch/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **是否必填** |
| **contractId** | 合同编号 | Y |
| **signParams** | 签署者及其签署参数列表，最多不超过10个签署者   |  |  |  | | --- | --- | --- | | **参数** | **说明** | **是否必填** | | **signaturePositions** | |  |  | | --- | --- | | **参数** | **说明** | | pageNum | 签名位置锁在页码 | | x | 签名位置的x坐标 | | y | 签名位置的y坐标 |   签署位置，JsonArray格式，每一项为一个json对象 | Y | | **signatureImageData** | 签名图片，用户指定的签名图片，Base64字符串 | N | | **signatureImageName** | 签名图片名称，签名图片名称，取自【2.2上传用户签名/印章图片】时设置的imageName，适用于多签章场景，可在签署时灵活使用不同的签章。 | N | | |

**示例：**

**{**

**"contractId"**: "13214235325435436457"**,**

**"signParams"**: **{**

**"签署人账号A"**: **{**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.1"**,**

**"y"**: "0.2"

**},**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.3"**,**

**"y"**: "0.4"

**}**

**],**

**"signatureImageData"**: ""**,**

**"signatureImageName"**: ""

**}**

**}**

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost"**: 906**,**

**"data"**: **{**

**"result"**: **{**

**"15868191451"**: **{**

**"errno"**: 0**,**

**"errmsg"**: ""

**},**

**"364966257@qq.com"**: **{**

**"errno"**: 0**,**

**"errmsg"**: ""

**}**

**}**

**},**

**"errmsg"**: ""**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.29、合同关键字批量签署（即批量自动签）

说明：按照关键字定位签署合同，可直接用于无感知的快捷签署。适用于同一份合同内的多个签署者同时签署，同一个签署者使用同一张签名/印章图片。

此接口要求签署者必须拥有自己的数字证书。可以无需先调用“添加签署者”接口与“设置合同签署参数”接口。

Request Path：**[server-host]**/contract/sign/keywords/batch/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |
| --- | --- | --- |
| **参数** | **说明** | **是否必填** |
| **contractId** | 合同编号 | Y |
| **signParams** | 签署者及其签署参数列表，最多不超过10个签署者，使用key:value的方式进行组装对象，key为签署者账号，value为签署参数，类型是JSONObejct，value参数如下：   |  |  |  | | --- | --- | --- | | **参数** | **说明** | **是否必填** | | **keywords** | 签署位置，JsonArray格式，每一项为一个json对象 | Y | | **align** | 对齐方式，枚举值如下：  lt 左上角对齐  lc 左中对齐  lb 左下对齐  ct 中上对齐  cc 正中对齐  cb 中下对齐  rt 右上角对齐  rc 右中对齐  rb 右下对齐 | N | | **signatureImageData** | 签名图片，用户指定的签名图片，Base64字符串 | N | | **signatureImageName** | 签名图片名称，签名图片名称，取自【2.2上传用户签名/印章图片】时设置的imageName，适用于多签章场景，可在签署时灵活使用不同的签章。 | N | | |

**示例：**

**{**

**"contractId"**: "150347901101000001"**,**

**"signParams"**: **{**

**"13456833929"**: **{**

**"keywords"**: **[**

"签章1"**,**

"签章2"

**],**

**"align"**: "cc"**,**

**"signatureImageData"**: ""**,**

**"signatureImageName"**: ""

**}**

**}**

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost"**: 2117**,**

**"data"**: **{**

**"result"**: **{**

**"iwanttosuccess4@bestsign.cn"**: **{**

**"errno"**: 0**,**

**"errmsg"**: ""

**},**

**"371121199207110833"**: **{**

**"errno"**: 0**,**

**"errmsg"**: ""

**}**

**}**

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.30、WPS专用签署合同

说明：签署合同，可直接用于无感知的快捷签署。此接口要求签署者必须拥有自己的数字证书。可以无需先调用“添加签署者”接口与“设置合同签署参数”接口。

由于wps的pdf文件，存在裁剪区域CropBox，我们正常的签署合成，是按照MediaBox来做的，导致签署位置定位不准确。此接口为WPS专用签署接口。

Request Path：**[server-host]**/contract/sign/cert/wps/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | |
| **contractId** | 合同编号 | Y |  |  | | |
| **signer** | 签署者 | Y |  | 即签署者的account | | |
| **signaturePositions** | 指定的签署位置，json array格式 | Y |  | 每一项为一个json对象 | | |
|  |  | 最大长度 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| y | 纵坐标，同上 | 10 |
| pageNum | 页码。1开始，不能超过实际页数 | 3 |
| rptPageNums | 当前位置的签名需要复制到的目标页码列表。该参数用于控制是否将当前位置的签名复制到其他页。参数启用之后，当前页的签名会同时出现在其他页的相同位置。参数值格式如下： 多个页码使用英文逗号(,)分隔。 参数不存在或者参数值为空表示不复制， 参数值为0表示复制到所有页， 参数值为其他表示复制到指定页。 参数值示例："0" "1,2,3,4,5,6" "1,3,5,7,9" |  |
| signatureImageData | 用户指定的签名图片，Base64字符串。表示允许每个位置使用不同的图片。允许为空，为空或者图片数据错误时使用外层的图片设置。 |  |
| signatureImageWidth | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。  目前如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。宽高为配套参数，非必填参数，此处不设置则使用外层的宽高设置。 |  |
| signatureImageHeight | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。  目前如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。宽高为配套参数，非必填参数，此处不设置则使用外层的宽高设置。 |  |
| **signatureImageName** | 签名图片名称 | N |  | 企业用户可以指定上传过的印章图片中的某一张作为本次签署的印章图片，取“上传用户签名/印章图片”接口设置的**imageName**的值 | | |
| **signatureImageData** | 签名图片 | N |  | 用户指定的签名图片，Base64字符串。允许为空，为空时使用用户默认的签名图片。  使用自动签的用户，可以越过“2.1、创建用户签名/印章图片”接口、或“2.2、上传用户签名/印章图片”接口，直接在此提供签名/印章图片即可。 | | |
| **signatureImageWidth** | 签名图片显示宽度 | N |  | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。 同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。 如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | | |
| **signatureImageHeight** | 签名图片显示高度 | N |  | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | | |

**示例：**

**{**

**"contractId"**: "cU34344234"**,**

**"signer"**: "u-1112343242"**,**

**"signatureImageData"**: "cU34344234"**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.05"**,**

**"y"**: "0.71"**,**

**"rptPageNums"**: "2,3,4,5,6,7"**,**

**"signatureImageData"**: "图片base64"**,**

**"signatureImageWidth"**: "100"**,**

**"signatureImageHeight"**: "70"

**},**

**{**

**"pageNum"**: "2"**,**

**"x"**: "0.05"**,**

**"y"**: "0.81"**,**

**"signatureImageData"**: "图片base64"**,**

**"signatureImageWidth"**: "100"**,**

**"signatureImageHeight"**: "70"

**}**

**]**

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.31、一次性签署合同

说明：本“一次性签署合同”接口，通常用于“信用卡领用合约”等一次性的签署场景。本接口属于无感知签署。

本接口由以下接口依次组合而成，实现以下接口功能功能：

*1.1、注册个人用户并申请证书*

*1.2、注册企业用户并申请证书*

*3.2、PDF文件添加元素*

*4.11、创建合同*

*4.2、签署合同（即自动签）*

*4.4、锁定并结束合同*

Request Path：**[server-host]**/contract/fidSign/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

URL示例：https://**[server-host]**/contract/fidSign/?developerId=xx&rtick=xxxxx&signType=rsa&sign=xxxxx

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **描述** |
| **businessId** | 业务编号 | N |  | 交行自定义的业务编号 |
| **title** | 合同标题 | Y | 50 | 交行自定义，可用于登录上上签后查询合同 |
| **description** | 合同描述 | N | 200 | 交行自定义 |
| **expireTime** | 合同签署的到期时间 | Y | 20 | 合同必须在指定的截止时间之前完成签署，一旦过期则此合同将无法被签署。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”；  交行自定义，unix时间戳可以取服务器时间；如果不传则默认为7天。 |
| **pdf** | pdf模版文件及元素 | Y | JSON格式 | 本次签署的合同模版文件，以及需要在模版上添加的元素   |  |  |  |  | | --- | --- | --- | --- | | **参数名** | **说明** | **是否必填** | **最大长度** | | **fid** | 模版文件编号，由交行调用上传文件接口获得，一期由上上签代为上传并告知交行此fid | Y |  | | **elements** | 要添加的元素集合JSONArray格式，即用户抄写声明的图片在此提交 | Y |  | |
| **senderAccount** | 发送者 | Y | JSON格式 | 发送者表示该合同是由谁发起的，可以固定一个account（开发者自己），也可以设置成与签署者为同一个account   |  |  |  |  | | --- | --- | --- | --- | | **参数名** | **说明** | **是否必填** | **最大长度** | | **account** | 用户唯一标识，交行自定义，可以用证件号码 | Y |  | | **mail** | 邮箱 | N | 50 | | **mobile** | 手机 | N | 20 | | **name** | 用户名称 | Y | 100 | | **userType** | 用户类型。 1 - 个人 | Y | 1 | |
| **signerAccounts** | 签署者 | Y | JSON格式 | 签署者即每一个参与合同签署的人，支持一次性传多个签署者   |  |  |  |  | | --- | --- | --- | --- | | **参数名** | **说明** | **是否必填** | **最大长度** | | **account** | 用户唯一标识，交行自定义，可以用证件号码 | Y |  | | **mail** | 邮箱 | N | 50 | | **mobile** | 手机 | N | 20 | | **name** | 用户名称 | Y | 200 | | **userType** | 用户类型。 1 - 个人 | Y | 1 | | **credential** | 证件信息, json格式，具体字段参见设置证件接口 | Y |  | | **signatureImageData** | 指定的签名图片，base64字符串 | Y |  | | **signatureImageWidth** | 签名图片宽度 | N |  | | **signatureImageHeight** | 签名图片高度 | N |  | | **signaturePositions** | 签名图片所在位置，JSONArray格式 | Y |  | |

**Request Body示例**：

**{**

**"businessId"**: ""**,**

**"title"**: ""**,**

**"description"**: ""**,**

**"expireTime"**: ""**,**

**"pdf"**: **{**

**"fid"**: ""**,**

**"elements"**: **[**

**{**

**"pageNum"**: "声明图片所在页码，从1开始，不能超过实际页数，必填"**,**

**"x"**: "声明图片的横坐标，取值0到1之间的小数，必填"**,**

**"y"**: "声明图片的纵坐标，取值0到1之间的小数，必填"**,**

**"type"**: "填image，必填"**,**

**"value"**: "为base64编码后的声明图片内容"

**}**

**]**

**},**

**"senderAccount"**: **{**

**"account"**: ""**,**

**"name"**: ""**,**

**"userType"**: ""**,**

**"mail"**: ""**,**

**"mobile"**: ""

**},**

**"signerAccounts"**: **[**

**{**

**"account"**: ""**,**

**"name"**: ""**,**

**"userType"**: ""**,**

**"mail"**: ""**,**

**"mobile"**: ""**,**

**"credential"**: **{**

**"identity"**: "证件号码，必填"**,**

**"identityType"**: "证件类型，0-居民身份证，1-护照，B-港澳居民往来内地通行证，C-台湾居民来往大陆通行证，E-户口簿，F-临时居民身份证，必填"**,**

**"contactMail"**: "联系邮箱，非必填"**,**

**"contactMobile"**: "联系手机号，必填"**,**

**"province"**: "省，非必填"**,**

**"city"**: "市，非必填"**,**

**"address"**: "地址，非必填"

**},**

**"signatureImageData"**: "图片的base64字符串"**,**

**"signatureImageWidth"**: "图片的宽度，可以不指定"**,**

**"signatureImageHeight"**: "图片的宽度，可以不指定"**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "签名图片所在页码，从1开始，不能超过实际页数，必填"**,**

**"x"**: "签名图片的纵坐标，取值0到1之间的小数，必填"**,**

**"y"**: "签名图片的横坐标，取值0到1之间的小数，必填"

**}**

**]**

**}**

**]**

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **contractId** | 合同ID |
| **businessId** | 业务编号 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"contractId"**: "150993706101000005"**,**

**"businessId"**: "JTYH-18736173"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.32、上传合同文件到公有云

说明：用于混合云模式的客户将已签署完成的合同文件上传到上上签公有云。

Request Path：

测试环境：https://openapi.bestsign.info/openapi/v2/contract/uploadContent/

生产环境：https://openapi**2**.bestsign.cn/openapi/v2/contract/uploadContent/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **contractId** | 合同编号 | Y |  |  |
| **fdata** | 文件内容 | Y |  | base64编码格式 |
| **isAllowWebDownload** | 是否允许被上上签官网注册用户查看下载 | N | 1 | 枚举值0/1，不带此参数则默认为1。  当签署者到上上签官网注册实名认证，并登录上上签后，根据此参数控制：  0-不显示在合同列表中，无法被查看、下载  1-显示在合同列表中，可以被查看、下载 |
| **hotStoragePeriod** | 热存周期 | N |  | 此参数是合同文件在热存储中保留的时间长度，单位为秒。保存在热存储中的合同数据，自合同结束时间算起，超过此此参数设定时长的合同文件，会转移到冷存储中。计算示例：如保存365天，则值为3600\*24\*365=31536000。参数可为空，为空时默认值为1年（31536000）。取值范围为3600（1小时）~157680000（5年） |

**示例：**

**{**

**"contractId"**: "132198432"**,**

**"isAllowWebDownload"**: "1"**,**

**"fdata"**: "xxxxxx"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数：无

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 4.33、锁定并结束合同（自定义证书）

说明：当所有签署者都签署完毕后，使用此接口添加上上签系统的锁定签名，并且结束整个合同签署过程，将合同状态更新为“已完成”，不允许任何人再进行签署。只有结束的合同才能提供公证服务。本接口允许使用指定账号的证书进行锁定。

Request Path：**[server-host]**/storage/contract/lock/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **contractId** | 合同编号 | Y |  |
| **signerAccount** | 账号 | N | 建议使用开发者自己、或者关键签署者的账号 |

**示例：**

**{**

**"contractId"**: "1703465339807805447"**,**

**"signerAccount"**: "mlimd@163.com"

**}**

**响应（RESPONSE）：**errno返回0则为正常，errno返回错误码则为异常，此时观察具体的errmsg具体分析。errno与errmsg请参阅本文《四、附录：返回信息编号列表》，返回信息编号列表持续更新。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.34、上传合同签署照片

说明：

此接口将需要体现在合同附页上的图片（如已通过人脸比对验证的人脸照片）上传，以便在合同签署结束后在附页中体现此图片。如果不调用此接口，则合同附页上不会体现图片。此接口必须在“签署合同”接口（手动签或自动签）之前调用。

Request Path：[**server**-host]/contract/uploadSignPhoto/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **contractId** | 合同编号 | Y |  |  |
| **account** | 用户唯一标识 | Y |  | 必须要指定一个用户唯一标识作为操作者 |
| **imagedata** | 文件数据，base64编码 | Y |  | 即需要体现在附页上的照片 |

**示例：**

**{**

"contractId"**:** "1741075002964312072"**,**

"account"**:** "u-1112343242"**,**

"imageData"**:** "......................."

**}**

**响应（RESPONSE）：**

正常示例：

**{**

"errno"**:** 0**,**

"data"**:** **{},**

"errmsg"**:** ""

**}**

异常示例：

**{**

"errno"**:** 240012**,**

"data"**:** **null,**

"errmsg"**:** "check sign wrong "

**}**

### 4.35、ISV创建合同

contract/create/{docType}

|  |  |
| --- | --- |
| **Method** | createByDocType |
| **请求类型** | POST |
| **说明** | 创建合同，会按照ISV模式进行统计与扣量，当充值量已用完时，仍可以创建合同，但是签署该合同时会提示无法签署，必须及时充值才可以继续签署 |
| **其他URL参数** | 无 |
| **需要参与签名的其他URL参数** | 无 |
| **REQUEST BODY** | docType, 0 :ToB, 1: ToC如果docType不传默认就统计为ToB合同   | **参数名** | **说明** | **是否必填** | **最大长度** |  | | --- | --- | --- | --- | --- | | **account** | 用户唯一标识 | Y |  | 必须是在开放平台设置过的enduser账号 | | **fid** | 文件id | Y |  |  | | **expireTime** | 合同过期时间 | N |  | 如“1490672595”,秒级的unix时间戳；默认过期天数是7天 | | **title** | 合同标题 | Y | 100 |  | | **description** | 合同描述 | N | 200 |  | |
| **RESPONSE DATA** | | **参数名** | **说明** | | --- | --- | | **contractId** | 合同编号 | |
| **示例** | Request Body：  {      "account": "u-1112343242",      "fid":"f-8b0351057dd",      "expireTime":"7",      "title":"demo",      "description":"a demo contract"  }    Response:  {      "errno":0,      "errmsg":"",      "data":{          "contractId":"c1343423222",      }  } |
|  |  |

### 4.36、带短信校验的自动签

说明：签署合同，可直接用于无感知的快捷签署。此接口要求签署者必须拥有自己的数字证书，提交手机号码与短信验证码，如验证码校验通过则完成签署，否则签署失败需重试。

使用此接口的流程为：“创建合同（接口4.1或接口4.11）——添加签署者（接口4.12或接口4.13）——发送短信验证码（接口6.1）——带短信校验的自动签（本接口）”。

Request Path：**[server-host]**/storage/contract/sign/cert2/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | |
| **contractId** | 合同编号 | Y |  |  | | |
| **signer** | 签署者 | Y |  | 即签署者的account | | |
| **signaturePositions** | 指定的签署位置，json array格式 | Y |  | 每一项为一个json对象 | | |
|  |  | 最大长度 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| y | 纵坐标，同上 | 10 |
| pageNum | 页码。1开始，不能超过实际页数 | 3 |
| rptPageNums | 当前位置的签名需要复制到的目标页码列表。该参数用于控制是否将当前位置的签名复制到其他页。参数启用之后，当前页的签名会同时出现在其他页的相同位置。参数值格式如下： 多个页码使用英文逗号(,)分隔。 参数不存在或者参数值为空表示不复制， 参数值为0表示复制到所有页， 参数值为其他表示复制到指定页。 参数值示例："0" "1,2,3,4,5,6" "1,3,5,7,9" |  |
| **signatureImageName** | 签名图片名称 | N |  | 企业用户可以指定上传过的印章图片中的某一张作为本次签署的印章图片，取“上传用户签名/印章图片”接口设置的**imageName**的值 | | |
| **signatureImageData** | 签名图片 | N |  | 用户指定的签名图片，Base64字符串。允许为空，为空时使用用户默认的签名图片。  使用自动签的用户，可以越过“2.1、创建用户签名/印章图片”接口、或“2.2、上传用户签名/印章图片”接口，直接在此提供签名/印章图片即可。 | | |
| **signatureImageWidth** | 签名图片显示宽度 | N |  | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。 如果有一个为空，则按照同时为空处理，即使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。 如果同时有值，则图片按照两个宽高值拉伸填充，效果可能会产生变形。 | | |
| **signatureImageHeight** | 签名图片显示高度 | N |  | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。 如果有一个为空，则按照同时为空处理，即使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。 如果同时有值，则图片按照两个宽高值拉伸填充，效果可能会产生变形。 | | |
| **sendTarget** | 需校验的手机号码 | Y |  | 使用“6.1、发送签署合同验证码”接口提交的手机号码 | | |
| **vcode** | 需校验的短信验证码 | Y | 6 | 使用“6.1、发送签署合同验证码”接口后得到的短信验证码,6位数字 | | |
| **loginIp** |  |  |  |  | | |
| **signIp** |  |  |  |  | | |
| **loginTime** |  |  |  |  | | |

**示例：**

**{**

**"contractId"**: "cU34344234"**,**

**"signer"**: "u-1112343242"**,**

**"signatureImageData"**: "cU34344234"**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.05"**,**

**"y"**: "0.71"**,**

**"rptPageNums"**: "2,3,4,5,6,7"

**},**

**{**

**"pageNum"**: "2"**,**

**"x"**: "0.05"**,**

**"y"**: "0.81"

**}**

**]**

**"sendTarget"**: "13857410069"**,**

**"vcode"**: "665423"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

常见的验证码错误有“验证码校验失败”和“验证码过期/不存在”。

验证码校验失败：

**{**

**"errno"**: 240016**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "failed to verify vcode"

**}**

验证码过期/不存在：

**{**

**"errno"**: 240020**,**

**"cost": 233,**

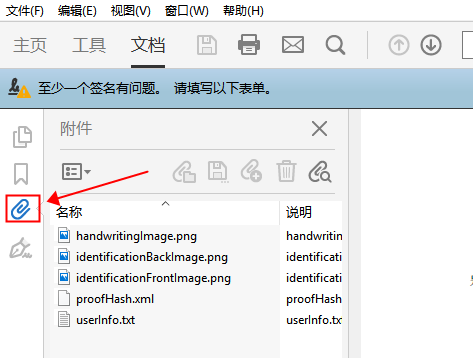
**"data"**: null**,**

**"errmsg"**: "vcode not exists"

**}**

### 4.37、合同PDF文件上添加附件

说明：

* 在合同PDF文件上添加附件，添加附件后的合同文件，用Adobe阅读器打开查看的效果如图所示：。
* 本接口一次只能添加一个附件，只能在合同未签署之前（即合同PDF文件上还未有任何签名之前）添加，否则会导致签名无效，文件最大不得超过10MB。

Request Path：**[server-host]**/dist/contract/pdfAddAttachment/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **contractId** | 合同编号 | Y |  |
| **fname** | 附件名称 | Y |  |
| **fdata** | 附件文件，base64编码字符串 | Y | 文件最大不得超过10MB |
| **fdescription** | 附件描述 | Y |  |

**示例：**

**{**

**"contractId"**: "149888108901000020"**,**

**"fname"**: "test.png"**,**

**"fdata"**: "iVBORw0KGgoAAAANSUhEUgAAAM0AAADNCAYAAAAbvPRpAACAAElEQV......"**,**

**"fdescription"**: "description"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

## 5、文件处理工具

### 5.1、PDF关键字查找位置

说明：

在PDF文件上查找关键字的位置坐标，返回坐标数组。开发者可以根据实际业务场景，选择使用所有坐标调用“自动签”或者“手动签”，也可以选择使用其中的个别坐标调用“自动签”或者“手动签”。

Request Path：[**server-host**]/pdf/findKeywordPositions/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **keyword** | 关键字 | Y |  | 需要查找的关键字 |
| **pdfData** | 文件数据，base64编码 | Y |  | 需要查找的PDF文件，文件内容的Base64字符串。例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fdata =Base64.encodeBase64String(bdata); |

**示例：**

**{**

"keyword"**:** "张三"**,**

"pdfData"**:** "file\_base64"

**}**

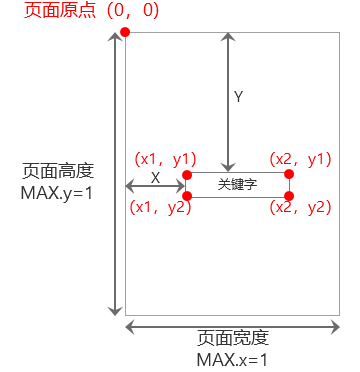
**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |  |
| --- | --- | --- |
| **参数名** | **说明** | |
| **positions** | 关键字位置列表，json array格式，4个数可以组成关键字四个角的位置 | |
| **参数名** | **说明** |
| **pageNum** | 关键字所在页码 |
| **x1** | 第一个x坐标, 用百分比表示在页面上的比例，取值0.0~1.0 |
| **x2** | 第二个x坐标, 用百分比表示在页面上的比例，取值0.0~1.0 |
| **y1** | 第一个y坐标, 用百分比表示在页面上的比例，取值0.0~1.0 |
| **y2** | 第二个y坐标, 用百分比表示在页面上的比例，取值0.0~1.0 |

正常示例：

**{**

图解

**"errno"**: 0**,**

**"cost"**: 49**,**

**"data"**: **{**

**"positions"**: **[**

**{**

**"y1"**: "0.7139"**,**

**"x1"**: "0.1814"**,**

**"y2"**: "0.7231"**,**

**"x2"**: "0.257"**,**

**"pageNum"**: "5"

**}**

**]**

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 5.2、获取PDF文件预览图片

说明：

获取PDF文件的预览图片，输入一个PDF，返回PDF每一页的预览图片集合。

Request Path：[**server-host**]/pdf/preview/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **pdfData** | 文件数据，base64编码 | Y | 5000000 | 例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fdata =Base64.encodeBase64String(bdata); |
| **dpi** | 图片清晰度 | Y | 3 | 预览图片清晰度，枚举值：96-低清（默认），120-普清，160-高清，240-超清 |

**示例：**

**{**

"pdfData"**:** "file\_base64"**,**

"dpi"**:** "96"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：image（byte[]）

正常示例：

正常响应返回文件流，文件流包含了所有pdf页码对应的图片的base64字符串集合。按照页码顺序存放，通过换行符来区分每一页图片。解析示例如下：InputStreaminputStream**=**response**.**body**().**byteStream**();**BufferedReaderbufferedReader**=**newBufferedReader**(**newInputStreamReader**(**inputStream**));**Stringstr**=null;**intpage**=**1**;**Stringdir**=**System**.**currentTimeMillis**()+**""**;**while**((**str**=**bufferedReader**.**readLine**())!=null){**

//返回结果包含errno，标识图片转换失败if(str.contains("errno")){

System**.**out**.**println**(**str**);**break**;**

**}**System**.**out**.**println**(**page**+**".jpg base64:"**+**Base64**.**getDecoder**().**decode**(**str**));**page**++;**

**}**//closebufferedReader.close();inputStream.close();

异常示例：

**{**

**"errno"**: 240006,

**"cost": 233**,

**"data"**: null,

**"errmsg"**: " no pdfData "

**}**

## 6、通知服务

### 6.1、发送签署合同验证码

说明：向指定手机号发送指定合同的签署验证码，用于【6.2 校验签署合同验证码】。

Request Path：[**server-host**]/contract/sendSignVCode/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **contractId** | 合同编号 | Y | 20 |
| **account** | 指定给哪个用户发验证码，用户必须是合同的签署者 | Y | 64 |
| **sendTarget** | 验证码发送目标手机号码，设置此项后，会更新掉【4.5 设置合同签署参数】接口设置vcodeMobile参数 | Y | 20 |
| **sendType** | 发送类型，目前支持：sms（短信）, voiceSms（语音）。以后可能支持mail | Y | 10 |

**示例：**

**{**

"contractId"**:** "1809032932740825699"**,**

"account"**:** "13456833929"**,**

"sendTarget"**:** "13456833929"**,**

"sendType"**:** "sms"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 6.2、校验签署合同验证码

说明：校验【6.1 发送签署合同验证码】发出的验证码。

Request Path：[**server-host**]/contract/verifySignVCode/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **contractId** | 合同编号 | Y | 20 |
| **account** | 指定给哪个用户发验证码，用户必须是合同的签署者 | Y | 64 |
| **vcode** | 6位数字验证码 | Y | 6 |

**示例：**

**{**

"contractId"**:** "cU34344234"**,**

"account"**:** "13867410069"**,**

"vcode"**:** "431225"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| result | 验证码校验结果；  1：校验通过；  0：校验不通过；  空：长度或字符等不规范； |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"result"**: "1"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 6.3、生成短链接

说明：传入长链接，获得上上签域名的短链接。短链接有效期固定为半小时。

Request Path：[**server-host**]/notice/shorturl/create/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **longUrl** | 长链接 | Y | 20 |
| **expireTime** | 短链接的到期时间，unix时间戳格式（秒数）。超过此时间则无法打开链接页面，需要获取新的签署合同链接。  注：并不是合同的可签署到期时间，只是此签署链接的有效期。  由于返回的是短链接，如果没有设置则默认是30分钟，如果设置了以设置的为准，但有效期最大不能超过7天。 | N | 20 |

**示例：**

**{**

**"longUrl"**: "http://baidu.com"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **shortUrl** | 短链接 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"shortUrl"**: "https://bestsign.cn/123456"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 6.4、发送通知短信

说明：

* 传入签署链接与手机号码，由上上签向指定手机号码发送签署链接的通知短信。
* 由于短信会产生费用，请在使用此功能前，先与上上签确定商务事宜，并通知上上签进行开发者的通知短信模版配置。

Request Path：[**server-host**]/notice/send/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **bizType** | 发送业务类型，“sign”表示签署合同。暂时只支持“sign”。 | Y | 20 |
| **channel** | 发送方式，暂时只支持支持“sms”。 | Y |  |
| **target** | 发送目标，**channel**“sms”为国内手机号码 | Y |  |
| **content** | 发送内容：jsonObject，根据实际业务的模板制定 | Y |  |

**示例：**

**{**

**"bizType"**: "sign"**,**

**"channel"**: "sms"**,**

**"target"**: "15757154266"**,**

**"content"**: **{**

**"shortUrl"**: "http://localhost:8080/s/12ade"

**}**

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **shortUrl** | 短链接 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

## 7、自定义合同编号的本地demo

此服务的功能有定制的本地demo代码实现。开发者可以通过此demo来满足自定义合同编号的功能，开发者可以将demo修改后使用，也可以参考demo的方法自行实现。

使用流程如下：

流程说明（必读）：

1. 获取“上上签合同编号”

调用接口“4.1、上传并创建合同”，从返回参数中取得上上签的合同编号；

1. 建立合同编号映射

调用demo方法“**aliasReg**”（或者调用接口“[**server-host**]/alias/reg/”）将取得的“上上签合同编号”与开发者自己的“自定义合同编号”进行映射，此映射关系保存在上上签；

1. 以“自定义合同编号”进行操作

建立好映射关系后，开发者就可以调用demo里的方法，直接使用“自定义合同编号”进行合同的签署、下载等操作，具体的方法请查阅下文“二）、本地demo方法列表”；

1. 查询“上上签合同编号”

当建立好映射关系后，开发者可以调用demo方法“**aliasGetInfo**”（或者调用接口“[**server-host**]/alias/getInfo/”）通过“自定义合同编号”来查询“上上签合同编号”，以便使用demo中未包含的其他功能。

### 一）、合同编号转换的通用方法

在这个本地demo的所有方法中，因涉及到“自定义合同编号”与“上上签合同编号”的映射转换，因此上上签提供了公共的映射合同编号的方法。

#### 1）、建立合同编号映射

此方法用于建立“自定义合同编号”与“上上签合同编号”之间的映射关系，该映射保存在上上签。

在本地demo的方法中如下：

private void **aliasReg(**String customContractId**,** String bestSignContractId**,** String aliasType**)** **throws** Exception;

另外，此方法也有独立的https接口，可以通过https直接调用，https接口如下：

Request Path：[**server-host**]/alias/reg/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **customId** | 自定义合同编号 | Y | 30 |
| **bestsignId** | 上上签的合同编号，通过接口“4.1、创建合同”取得 | Y |  |
| **type** | 业务类型，填“contractId” | Y |  |

**示例：**

**{**

**"customId"**: "ttt11111111112"**,**

**"bestsignId"**: "1111111111111111111"**,**

**"type"**: "contractId"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 2）、查询合同编号

使用此方法，通过“自定义合同编号”获取已经建立映射的“上上签合同编号”。

在本地demo的方法中如下：

private JSONObject **aliasGetInfo(**String customId**,** String aliasType**)** **throws** Exception;

另外，此方法也有独立的https接口，可以通过https直接调用，https接口如下：

Request Path：[**server-host**]/alias/getInfo/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **customId** | 自定义合同编号 | Y | 30 |
| **type** | 业务类型，填“contractId” | Y |  |

**示例：**

**{**

**"customId"**: "ttt11111111112"**,**

**"type"**: "contractId"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **bestsignId** | 上上签合同编号 |
| **customId** | 自定义合同编号 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"bestsignId"**: "1111111111111111111"**,**

**"customId"**: "ttt11111111112"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 二）、本地demo方法列表

此demo包含的功能有：

|  |  |
| --- | --- |
|  | **功能** |
| 1 | 单个添加签署者 |
| 2 | 批量添加签署者 |
| 3 | 设置合同签署参数 |
| 4 | 签署合同 |
| 5 | 获取合同手动签署页URL（长） |
| 6 | 锁定结束合同 |
| 7 | 下载合同文件 |
| 8 | 生成短链接 |
| 9 | 拒绝签署 |

#### 7.1、通过URL创建合同

说明：传入自定义合同编号、合同文件的URL，创建合同。自定义合同编号与上上签合同编号的映射关系由上上签保存。

注：如果想要确认文件是否上传成功，除了根据本接口返回的response判断外，还可以通过“2）、查询合同编号”功能查询上上签的合同编号，如果能查到合同编号说明已经文件已经上传成功并创建合同了。

**方法：**

public void **contractCreate(**String customContractId**,** String account**,** String pdfFileUrl**,** String fileName**,** String expireTime**,** String title**,** String description**)** **throws** Exception;

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **customContractId** | 自定义合同编号 | Y | 30 | 由开发者自行定义的合同编号 |
| **account** | 用户唯一标识 | Y |  | 必须要指定一个用户唯一标识作为合同的创建者（建议统一为开发者的account，或者需要进行分类统计的某些account，以便后期按此account进行统计） |
| **pdfFileUrl** | 合同文件URL | Y |  | 合同文件在云存储上的URL，上上签通过此URL直接下载合同文件，该文件必须是pdf格式，否则接口会报错 |
| **fileName** | 文件名称 | Y |  | 合同文件的文件名 |
| **expireTime** | 合同签署的到期时间 | Y |  | 合同必须在指定的到期时间之前完成签署，一旦过期则此合同将无法被签署。格式为秒级的unix时间戳，如需要在“2017/11/23 14:47:43”到期则设置为“1511419663”； |
| **title** | 合同标题 | Y | 50 | 客户可以将自己的业务订单编号、合同标题、合同类型等放此处 |
| **description** | 合同描述 | N | 200 | 对此合同的说明描述 |

**示例：**

**{**

**"order"**: "自定义合同编号"**,**

**"account"**: "mlimd@163.com"**,**

**"url"**: "合同文件url"**,**

**"fid"**: "1702695409383636995"**,**

**"expireTime"**: "1513311132"**,**

**"title"**: "测试"**,**

**"description"**: "postman测试合同"

**}**

**响应（RESPONSE）：**errno返回0则为正常，errno返回错误码则为异常，此时观察具体的errmsg具体分析。errno与errmsg请参阅本文《四、附录：返回信息编号列表》，返回信息编号列表持续更新。

RESPONSE DATA参数如下：无

正常示例：

**{**

**"errno"**: 0**,**

**"data"**: **{},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240006**,**

**"data"**: null**,**

**"errmsg"**: "bad contractId"

**}**

#### 7.2、签署合同

说明：

①、签署合同，可直接用于无感知的快捷签署，此接口要求签署者必须拥有自己的数字证书。

②、调用此接口签署合同，可以不需要先调用“7.9、单个添加合同签署者”、“7.10、批量添加合同签署者”、“7.11、设置合同签署参数”这3个接口，只要调用此接口一次就可以完成签署。

本方法由“4.2、签署合同”封装而成。

方法：

public String **contractSignCert(**String customContractId**,** String signer**,** String signatureImageData**,** JSONArray signaturePositions**)** **throws** Exception;

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | |
| **customContractId** | 自定义合同编号 | Y |  | 由开发者自行定义的合同编号 | | |
| **signer** | 签署者 | Y |  | 即签署者的account | | |
| **signaturePositions** | 指定的签署位置，json array格式 | Y |  | 每一项为一个json对象 | | |
|  |  | 最大长度 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| y | 纵坐标，同上 | 10 |
| pageNum | 页码。1开始 | 3 |
| **signatureImageData** | 签名图片 | N |  | 用户指定的签名图片，Base64字符串。允许为空，为空时使用用户默认的签名图片。  使用自动签的用户，可以越过“2.1、创建用户签名/印章图片”接口、或“2.2、上传用户签名/印章图片”接口，直接在此提供签名/印章图片即可。 | | |

**示例：**

**{**

**"customContractId"**: "自定义合同编号"**,**

**"signer"**: "u-1112343242"**,**

**"signatureImageData"**: "图片的base64字符串"**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.05"**,**

**"y"**: "0.01"

**},**

**{**

**"pageNum"**: "2"**,**

**"x"**: "0.05"**,**

**"y"**: "0.01"

**}**

**]**

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 7.4、锁定并结束合同

说明：当所有签署者都签署完毕后，使用此接口添加上上签系统的锁定签名，并且结束整个合同签署过程，将合同状态更新为“已完成”，不允许任何人再进行签署。本方法由“4.5、锁定并结束合同”封装而成。

方法：

public String **contractLock(**String customContractId**)** **throws** Exception;

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **customContractId** | 自定义的合同编号 | Y |  |

**示例：**

**{**

**"customContractId"**: "1703466073594003458"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 7.5、下载合同文件

说明：通过合同编号下载对应的合同文件。本方法由“4.9、下载合同文件”封装而成。

方法：

public byte**[]** **contractDownload(**String customContractId**)** **throws** Exception;

需要参与签名的其他URL参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **customContractId** | 合同编号 | Y |  |

**请求参数（REQUEST BODY）：**无

**响应（RESPONSE）：**正常则返回pdf文件的文件流，errno返回错误码则为异常，此时观察具体的errmsg具体分析。errno与errmsg请参阅本文《四、附录：返回信息编号列表》，返回信息编号列表持续更新。

正常示例：

**返回文件流**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 7.6、生成短链接

说明：传入长链接，获得上上签域名的短链接。短链接有效期固定为半小时。本方法由“6.3、生成短链接”封装而成。

方法：

public String **noticeShorturlCreate(**String longUrl**)** **throws** Exception;

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **longUrl** | 长链接 | Y | 20 |

**示例：**

**{**

**"longUrl"**: "http://baidu.com"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **shortUrl** | 短链接 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"shortUrl"**: "https://bestsign.cn/123456"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 7.8、拒绝签署

说明：当某个签署者对合同表示有问题，不想继续签署时，或者当发起人发现有问题需要取消合同时，使用此接口结束合同。拒绝签署后此合同结束，其他未签署的签署者无法继续操作。本方法由“4.4、拒绝签署”封装而成。

方法：

public String **contractRefuse(**String customContractId**,** String signer**,** String refuseReason**)** **throws** Exception;

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |
| **customContractId** | 自定义的合同编号 | Y |  |
| **signer** | 签署者的account | Y |  |
| **refuseReason** | 拒绝原因 | Y | 200 |

**示例：**

**{**

**"customContractId"**: "1703466073594003458"**,**

**"signer"**: "mlimd@163.com"**,**

**"refuseReason"**: "不要吃兔兔~"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 7.9、单个添加签署者

说明：为已创建的合同添加签署者，添加签署者之后的合同正式发送。本方法由“4.12、单个添加签署者”封装而成。

方法：

public String **contractAddSigner(**String customContractId**,** String signer**)** **throws** Exception;

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **customContractId** | 自定义的合同编号 | Y | 创建合同时使用的自定义合同编号 |
| **signer** | 要添加的签署者account | Y |  |

**示例：**

**{**

**"customContractId"**: "17026986666669878"**,**

**"signer"**: "abc@163.com"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 7.10、批量添加签署者

说明：为已创建的合同批量添加签署者（即一次性添加多个签署者），添加签署者之后的合同正式发送。本方法由“4.13、批量添加签署者”封装而成。

方法：

public String **contractAddSigners(**String customContractId**,** JSONArray signers**)** **throws** Exception;

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **customContractId** | 自定义的合同编号 | Y | 创建合同时使用的自定义合同编号 |
| **signers** | 要添加的签署者account | Y | json array格式 |

**示例：**

**{**

**"customContractId"**: "1702698069495119878"**,**

**"signers"**: **[**

"13867410069"**,**

"13067825582"

**]**

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

返回各account添加成功与否情况。格式为"account":"1/0"，成功为"1"，失败为"0"。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"13067825582"**: "1"**,**

**"13867410069"**: "1"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 7.11、设置合同签署参数

说明：为签署者设置合同签署参数，包括签署位置的坐标（如坐标为设置则以签署接口为准）、使用的签名/印章图片等。本方法由“4.14、设置合同签署参数”封装而成。

方法：

public String **setSignerConfig(**String customContractId**,** String account**,** JSONArray signaturePositions**,** String isAllowChangeSignaturePosition**,** String returnUrl**,** String vcodeMobile**,** String isDrawSignatureImage**)** **throws** Exception;

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | | | **是否必填** | **默认值** | **最大长度** |
| **customContractId** | 自定义合同编号 | | | Y |  | 30 |
| **account** | 签署者 | | | Y |  |  |
| **signaturePositions** | 指定的默认签署位置，json array格式，每一项为一个json对象。如用手动签模式则此项必填，自动签可不填。  x,y坐标使用百分比，取值0.0 - 1.0。如果不指定此参数，表示由签署者自行在页面上选一个签署位置 | | | N |  |  |
| **参数名** | **说明** | **最大长度** |
| y | 纵坐标，同上 | 10 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| pageNum | 页码，从1开始，不能超过合同最大页数 | 3 |
| **isAllowChangeSignaturePosition** | 在有指定signaturePOSTion参数的情况下，是否允许拖动签名位置。取值1/0。（0：不允许，1：允许，下面都是这样的） | | | N | 1 | 1 |
| **returnUrl** | 手动签署时，当用户签署完成后，指定回跳的页面地址 | | | N |  |  |
| **vcodeMobile** | 手动签署用来接收短信验证码的手机号。  如手动签需要校验短信验证码则必填此项。  如不填写则手动签时不校验短信验证码。 | | | N |  | 20 |
| **isDrawSignatureImage** | 手动签署时是否手绘签名。取值0/1。  0手动签署时点击签名图片不会触发手写面板。 1手动签署时可以点击签名图片触发手写面板。 | | | N | 1 | 1 |

**示例：**

**{**

**"customContractId"**: "自定义合同编号"**,**

**"account"**: "u-34234234324"**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.05"**,**

**"y"**: "0.01"

**},**

**{**

**"pageNum"**: "2"**,**

**"x"**: "0.05"**,**

**"y"**: "0.01"

**}**

**],**

**"isAllowChangeSignaturePosition"**: "0"**,**

**"returnUrl"**: ""**,**

**"vcodeMobile"**: ""**,**

**"isDrawSignatureImage"**: "1"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

#### 7.12、获取手动签署URL

说明：获取合同手动签署的页面URL，可以根据设置的合同签署参数选择进行短信校验、手绘签名等操作，并在此页面完成签署。使用之前需要先调“7.9、单个添加签署者”与“7.11、设置合同签署参数”。

本方法由“4.3、发送合同”封装而成。

方法：

public String **getSignURL(**String customContractId**,** String account**,** String expireTime**,** String quality**)** **throws** Exception;

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **默认值** |
| **customContractId** | 自定义的合同编号 | Y |  |  |
| **account** | 指定给哪个用户看 | Y |  |  |
| **quality** | 预览图片质量，0-100，数值越高越清晰。 | N | 10 | 90 |
| **expireTime** | 签署链接的过期时间，unix时间戳格式（秒数）。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”。超过此时间则无法打开链接页面，需要获取新的签署合同链接。  注：并不是合同的可签署到期时间，只是此签署链接的有效期。最长不能超过当前时间的7天。 | N | 20 | 0 |
| **sid** | 开发者自定义的业务流水号（比如订单号之类），如有填写，则在异步推送签署结果时包含此业务流水号（拟增参数，暂时不支持） | N | 30 |  |

**示例：**

**{**

**"customContractId"**: "cU34344234"**,**

**"expireTime"**: "0"**,**

**"quality"**: "90"**,**

**"account"**: "13867410069"**,**

**"sid"**: "TBD192483219847321"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

手动签署页面的url，签署者在此页面上进行签署操作。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**http://tool.oschina.net/tools/json_format/Expanded.gif

**"url"**: "https://..........."

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

## 8、多文档合同服务

在某些特定的业务场景中，比如贷款购车时，需要同时签署“购车合同”、“征信调查授权书”、“抵押借款协议”以及其他辅助材料等，在这些场景下使用此功能，可以达到“校验一次验证码，同时签署多份合同文件”的效果。多文档合同服务是为这些特殊场景服务的，常规的单份合同的签约场景请使用单文档合同服务。

接口流程：

### 8.1、创建合同目录

说明：使用多文档签署功能，必须先创建一个合同目录，用来容纳需要同时签署的多份合同文件。同一个开发者创建的合同目录不能重名。

Request Path：[**server-host**]/catalog/create/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **senderAccount** | 用户唯一标识 | Y |  | 必须要指定一个用户唯一标识作为目录的创建者（建议统一为开发者的account，或者需要进行分类统计的某些account，以便后期按此account进行统计） |
| **expireTime** | 目录签署的截止时间 | Y |  | 合同必须在指定的截止时间之前完成签署，一旦过期则此合同将无法被签署。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”； |
| **catalogName** | 目录名称 | N | 128 | 可以将业务名称、自定义的流水号等等填写在此处，同一个开发者下的目录名称不能重复 |
| **description** | 目录描述 | N | 200 | 关于目录的描述 |

**示例：**

**{**

**"senderAccount"**: "mlimd@163.com",

**"expireTime"**: "1513311132",

**"catalogName"**: "测试",

**"description"**: "postman测试合同"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **catalogName** | 如果请求参数中没有填写则返回目录编号 |

正常示例：

**{**

**"errno"**: 0,

**"cost": 233**,

**"data"**: **{**

**" catalogName"**: "150993706101000005"

**}**,

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.2、向目录添加合同文件（根据fid或者fdata添加）

说明：创建好合同目录后，将需要同时签署的合同文件添加到目录中。已结束的目录不可再添加合同文件。要一份份单独的添加，支持采用“文件编号”和“PDF文件”2种方式，2种方式二选一即可：

* 通过“文件编号”添加——实现上传好合同文件，然后在此接口使用文件编号进行添加，用到的参数为“**fid**”，当填写了fid时系统只判断fid，忽略PDF文件方式的参数；
* 通过“PDF文件”添加——在此接口提交合同PDF文件（base64编码），用到的参数为“**fmd5，fname，ftype，fpages，fdata**”，使用文件方式不需要填写fid；

Request Path：**[server-host]**/catalog/uploadContract/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **senderAccount** | 用户唯一标识 | Y | 64 | 必须要指定一个用户唯一标识作为操作者 |
| **catalogName** | 目录名称 | Y |  |  |
| **fid** | 文件编号 | Y |  | 支持添加已经上传过的文件，如果是通过fid添加，则不需要参数“**fmd5，fname，ftype，fpages，fdata**”。 |
| **title** | 合同文件标题 | Y | 50 | 每个合同文件的标题，在同一个目录下的合同标题不得重复。一般是“XXX协议”、“XXX授权书”、“XXX” |
| **fmd5** | 文件md5值 | Y | 100 | 例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fmd5 = DigestUtils.md5Hex(bdata); |
| **fname** | 文件名 | Y | 50 | 文件名必须带上后缀名，例如“XXXX.pdf” |
| **ftype** | 文件类型 | Y | 10 | 仅支持PDF |
| **fpages** | 文件页数 | Y | 3 |  |
| **fdata** | 文件数据，base64编码 | Y | 5120000 | 例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fdata =Base64.encodeBase64String(bdata); |
| **expireTime** | 合同签署的到期时间 | Y | 20 | 合同必须在指定的到期时间之前完成签署，一旦过期则此合同将无法被签署。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”； |
| **description** | 合同描述 | N | 200 |  |
| **hotStoragePeriod** | 热存周期 | N |  | 此参数是合同文件在热存储中保留的时间长度，单位为秒。保存在热存储中的合同数据，自合同结束时间算起，超过此此参数设定时长的合同文件，会转移到冷存储中。计算示例：如保存365天，则值为3600\*24\*365=31536000。参数可为空，为空时默认值为1年（31536000）。取值范围为3600（1小时）~157680000（5年） |

**示例：**

**{**

**"senderAccount"**: "mlimd@163.com",

**"catalogName"**: "测试合同标题",

**"fid"**: "文件编号",

**"title"**: "合同标题",

**"fmd5"**: "文件的md5值",

**"ftype"**: "pdf",

**"fname"**: "滴滴电子发票.pdf",

**"fpages"**: "1",

**"fdata"**: "文件的base64字符串",

**"description"**: "测试合同描述",

**"expireTime"**: "1531242131"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **title** | 合同文件标题 |

正常示例：

**{**

**"errno"**: 0,

**"cost": 233**,

**"data"**: **{**

**"title"**: "XXX购车20171226"

**}**,

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.3、获取合同列表

说明：添加完合同文件后，使用此接口获取合同文件的列表，返回值中的“contractId”用于后续设置每个签署者的签署坐标等参数。

Request Path：[**server-host**]/catalog/getContracts/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |
| --- | --- | --- |
| **参数名** | **说明** | **是否必填** |
| **catalogName** | 目录名称 | Y |

**示例：**

**{**

**"catalogName"**: "测试"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |  |
| --- | --- | --- |
| **参数名** | **说明** | |
| **catalog** | 目录信息，json格式，包含以下参数： | |
| **参数名** | **说明** |
| **catalogId** | 目录ID |
| **developerId** | 开发者编号 |
| **userId** | 用户ID，无需关注 |
| **catalogName** | 目录名称，唯一标识 |
| **description** | 目录描述 |
| **status** | 目录状态，无需关注 |
| **isDel** | 是否被删除，无需关注 |
| **expireTime** | 目录签署的截止时间 |
| **sendTime** | 目录创建时间 |
| **finishTime** | 目录结束时间 |
| **contracts** | 合同文件列表，json格式，包含以下参数： | |
| **参数** | **说明** |
| **contractId** | 合同文件的合同编号，用于后续接口设置签署者的签署位置等 |
| **title** | 合同文件的标题 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost"**: 13**,**

**"data"**: **{**

**"catalog"**: **{**

**"catalogId"**: 8932921724828023000**,**

**"developerId"**: 2018635354264830500**,**

**"userId"**: 152948559601000000**,**

**"catalogName"**: "8月份"**,**

**"description"**: "测试合同目录"**,**

**"status"**: 2**,**

**"isDel"**: 0**,**

**"expireTime"**: 1544088944**,**

**"sendTime"**: "2018-08-23 11:43:14.0"**,**

**"finishTime"**: ""

**},**

**"contracts"**: **{**

**"3号"**: **{**

**"contractId"**: "153431606501000003"**,**

**"title"**: "3号"

**},**

**"1号"**: **{**

**"contractId"**: "153475715401000001"**,**

**"title"**: "1号"

**}**

**}**

**},**

**"errmsg"**: ""**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.4、获取多文档手动签署URL

说明：获取合同手动签署的页面URL，将此URL发送给指定用户，即可在此页面进行手动签操作完成签署。在此页面上实现“同时查看多份合同文件、只校验一次验证码，同时签署多份合同文件”的目的。查看合同时图片按合同编号排序。

请注意：本接口是签署接口，只能查看到需要签署的那几份合同，如果目录中有些合同不需要签署则不展示这些合同。

Request Path：[**server-host**]/catalog/send/

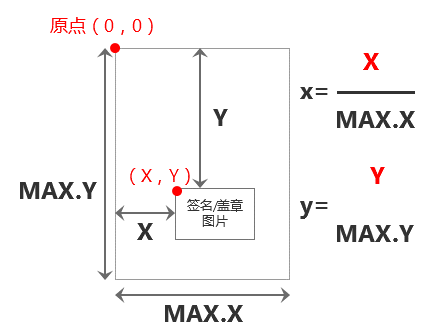
Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **默认值** |
| **catalogName** | 目录名称 | Y |  |  |
| **signerAccount** | 指定的签署者账号 | Y |  |  |
| **returnUrl** | 手动签署时，当用户签署完成后，签署页面指定回跳的页面地址，如果没填此参数则签署完成后跳转到合同预览页面。  如果returnurl带有“http”或“https”开头则判定h5跳转，如果returnurl不带“http”或“https”开头则判定为微信小程序跳转。 | N |  |  |
| **vcodeMobile** | 手动签署时指定接收手机验证码的手机号，如果需要采用手动签署页面校验短信验证码，则此项必填。  如果手动签不需要短信校验，则此项可不填。 | N |  | 20 |
| **dpi** | 预览图片清晰度，枚举值：96-低清（默认），120-普清，160-高清，240-超清 | N | 3 | 96 |
| **version** | 手动签的版本，默认为2，枚举值：  2-v2版本,  3-v3版本，UI优化，移动端手绘面板自动横屏 | N | 1 | 2 |
| **isDrawSignatureImage** | 手动签署时是否手绘签名。取枚举值0/1/2。  0点击签名图片不会触发手写面板（禁止手写）。  1点击签名图片能触发手写面板（既可手写也可使用默认签名）。  2强制必须手绘签名（只能手写不允许使用默认签名）。 | N | 1 | 1 |
| **readAll** | 枚举值0/1，默认为0  0-无需拖动到页面底部即可确认签署  1-必须拖动到页面底部，表示阅读完毕才能确认签署 | N |  |  |
| **signatureImageWidth** | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | N |  |  |
| **signatureImageHeight** | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | N |  |  |
| **isShowHandwrittenTime** | 枚举值0/1，默认为0  1-手写面板及签名图片上显示手写签名操作的时间  0-不显示该时间 | N | 1 | 0 |
| **isFaceAuth** | 是否使用刷脸签。1-刷脸签，使用刷脸校验；  0或其他-非刷脸签，使用短信校验或无校验； | N |  |  |
| **sid** | 流水号，开发者自定义，通常是开发者自己的业务流程中需要据此判断操作行为 | N |  |  |
| **contractParams** | 合同组，json格式。按每个合同文件的标题分组。 | | | |
| **参数名** | **说明** | **最大长度** | |
| 合同文件的标题 | 每个需要签署的合同文件的标题，即“获取合同列表”接口返回的title参数的值 |  | |
| signaturePositions | json数组格式，包含3个变量：pageNum、x、y |  | |
| pageNum | 页码，从1开始，不能超过合同最大页数 | 3 | |
| x | 横坐标，按页面尺寸的百分比计算，取值大于0小于1的小数，允许小数点。以左上角为原点 | 10 | |
| y | 纵坐标，同上 | 10 | |



**坐标像素换算公式：**

文档页面的左上角为坐标原点（0，0）

X：签名图片左上角离原点的实际X轴距离

Y：签名图片左上角离原点的实际Y轴距离

MAX\_x：文档页面的最大宽度

MAX\_y：文档页面的最大高度

X/MAX\_x=x表示换算后的x坐标比例x

Y/MAX\_y=y表示换算后的y坐标比例y

**示例：**

**{**

**"catalogName"**: "per1514302565867"**,**

**"vcodeMobile"**: "13888888888"**,**

**"returnUrl"**: "http://www.baidu.com"**,**

**"contractParams"**: **{**

**"合同一的标题"**: **{**

**"signaturePositions"**: **[**

**{**

**"x"**: "0.5"**,**

**"y"**: "0.5"**,**

**"pageNum"**: 1

**},**

**{**

**"x"**: "0.5"**,**

**"y"**: "0.5"**,**

**"pageNum"**: 2

**}**

**]**

**},**

**"合同二的标题"**: **{**

**"signaturePositions"**: **[**

**{**

**"x"**: "0.5"**,**

**"y"**: "0.5"**,**

**"pageNum"**: 1

**},**

**{**

**"x"**: "0.5"**,**

**"y"**: "0.5"**,**

**"pageNum"**: 2

**}**

**]**

**}**

**},**

**"signerAccount"**: "per1514302565867"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

手动签署页面的URL，签署者在此页面上进行签署操作。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"url"**: "https://..........."

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.5、获取预览合同URL

说明：获取合同预览的页面URL，在此页面预览合同内容。查看合同时图片按合同编号排序。

Request Path：[**server-host**]/catalog/getPreviewURL/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **默认值** |
| **catalogName** | 目录名称 | Y |  |  |
| **signerAccount** | 指定的签署者账号 | Y |  |  |
| **dpi** | 预览图片清晰度，枚举值：96-低清（默认），120-普清，160-高清，240-超清 | N | 3 | 96 |
| **expireTime** | 预览链接的过期时间，unix时间戳。超过此时间则无法预览合同，需要获取新的预览合同url。  注：并不是合同的可签署到期时间，只是此预览链接的有效期。 | N | 20 | 0 |

**示例：**

**{**

**"catalogName"**: "cU34344234",

**"expireTime"**: "0",

**"dpi"**: "120",

**"signerAccount"**: ""

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

预览合同的URL，签署者在此页面上进行合同预览操作。

正常示例：

**{**

**"errno"**: 0,

**"cost": 233**,

**"errmsg"**: "",

**"data"**: **{**

**"url"**: "https://..........."

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.6、签署合同（即自动签）

说明：签署合同，可直接用于无感知的快捷签署。此接口要求签署者必须拥有自己的数字证书。可以无需先调用“设置签署者参数”接口与“设置合同参数”接口。

由于自动签署不需要校验短信验证码，因此本接口为签署单份合同文件的接口相同。

Request Path：**[server-host]**/contract/sign/cert/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | |
| **contractId** | 合同编号 | Y |  |  | | |
| **signerAccount** | 签署者 | Y |  | 即签署者的account | | |
| **signaturePositions** | 指定的签署位置，json array格式 | Y |  | 每一项为一个json对象 | | |
| 参数 | 说明 | 最大长度 |
| x | 横坐标，按页面尺寸的百分比计算，取值为大于0小于1的小数。以左上角为原点 | 10 |
| y | 纵坐标，同上 | 10 |
| pageNum | 页码。1开始 | 3 |
| signatureImageData | 用户指定的签名图片，Base64字符串。表示允许每个位置使用不同的图片。允许为空，为空或者图片数据错误时使用外层的图片设置。 |  |
| signatureImageWidth | 本处位置使用的签名/印章图片在合同PDF上显示的宽度，填写DPI=72下的像素值，例如：想显示成300（宽）×200（高）则填300。宽高为配套参数，非必填参数，此处不设置则使用外层的宽高设置。 |  |
| signatureImageHeight | 本处位置使用的签名/印章图片在合同PDF上显示的宽度，填写DPI=72下的像素值，例如：想显示成300（宽）×200（高）则填200。宽高为配套参数，非必填参数，此处不设置则使用外层的宽高设置。 |  |
| **signatureImageData** | 签名图片 | N |  | 用户指定的签名图片，Base64字符串。允许为空，为空时使用用户默认的签名图片。  使用自动签的用户，可以越过“2.1、创建用户签名/印章图片”接口、或“2.2、上传用户签名/印章图片”接口，直接在此提供签名/印章图片即可。 | | |

**示例：**

**{**

**"contractId"**: "cU34344234",

**"signerAccount"**: "u-1112343242",

**"signatureImageData"**: "cU34344234",

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1",

**"x"**: "0.05",

**"y"**: "0.71"

**}**,

**{**

**"type"**: "date", //表示此位置为“签署日期”

**"pageNum"**: "2",

**"x"**: "0.05",

**"y"**: "0.81"

**}**

**]**

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.7、锁定并结束目录

说明：当一个合同目录下的所有签署者都签署完毕后，使用此接口添加上上签系统的锁定签名，并且结束整个合同签署过程，将合同状态更新为“已完成”，不允许任何人再进行签署。只有结束的合同才能提供公证服务。如果开发者需要下载签署完成的合同PDF文件给其他电子签约系统（比如资金方用的友商的电子签约系统）签署，那么请使用“8.8、结束目录”接口结束目录。

Request Path：[server-host]/catalog/lock/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **catalogName** | 目录名称 | Y |  | 可以将业务编号、自定义的流水号等等填写在此处，便于对接 |

**示例：**

**{**

**"catalogName"**: "测试"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| 每份合同文件的结果 | 如果正常则errno是0，如果异常则errno返回错误码 |

正常示例：

**{**

**"errno"**: 0,

**"cost"**: 359,

**"data"**: **{**

**"合同文件一的标题"**: **{**

**"errno"**: "0",

**"contractId"**: "151430629001000002",

**"errmsg"**: ""

**}**,

**"合同文件二的标题"**: **{**

**"errno"**: "0",

**"contractId"**: "151430629001000001",

**"errmsg"**: ""

**}**

**}**,

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.8、结束目录

说明：当一个合同目录下的所有签署者都签署完毕后，使用此接口结束整个合同签署过程，将合同状态更新为“已完成”，不允许任何人再进行签署，但不添加上上签的系统签名。只有结束的合同才能提供公证服务。如果开发者需要下载签署完成的合同PDF文件给其他电子签约系统（比如资金方用的友商的电子签约系统）签署，那么请使用本接口结束目录。

Request Path：[server-host]/catalog/end/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **catalogName** | 目录名称 | Y |  | 可以将业务编号、自定义的流水号等等填写在此处，便于对接 |

**示例：**

**{**

**"catalogName"**: "测试"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| 每份合同文件的结果 | 如果正常则errno是0，如果异常则errno返回错误码 |

正常示例：

**{**

**"errno"**: 0,

**"cost"**: 359,

**"data"**: **{**

**"合同文件一的标题"**: **{**

**"errno"**: "0",

**"contractId"**: "151430629001000002",

**"errmsg"**: ""

**}**,

**"合同文件二的标题"**: **{**

**"errno"**: "0",

**"contractId"**: "151430629001000001",

**"errmsg"**: ""

**}**

**}**,

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.9、生成目录的附页

说明：当目录结束后，可以调用此接口为目录生成一个附页文件，该附页文件是上上签提供的对合同从创建到签署完毕的整个签署过程的证明。一个目录仅对应一份附页文件。

Request Path：**[server-host]**/catalog/createAttachment

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **catalogName** | 目录名称 | Y |  |

**示例：**

**{**

**"catalogName"**: "测试"

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.10、下载目录附页文件

说明：用此接口可以下载附页文件。

Request Path：**[server-host]**/catalog/downloadAttachment

Method：GET

其他url参数：**catalogName**

需要参与签名的其他URL参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** |  |
| **catalogName** | 目录名称 | Y |  |

**请求参数（REQUEST BODY）：无**

**示例：**

**GET [**server-host**]/**catalog**/**downloadAttachment**/?**catalogName**=**d445f5432**&**developerId**=**d445f5432**&**rtick**=**17364899234**&**sign**=**a5334df21323**&**signType**=**rsa**&**contractId=150496133501000001

**响应（RESPONSE）：**

正常示例：

**返回文件流**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.11、向目录添加合同文件（根据contractId添加）

* 说明：创建好合同目录后，将需要同时签署的合同文件添加到目录中。已结束的目录不可再添加合同文件。要一份份单独的添加，本接口支持采用“contractId”的方式。

Request Path：[**server**-host]/catalog/addExistedContract/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **catalogName** | 目录名称 | Y |  | 已经创建的目录名称 |
| **contractId** | 合同编号 | Y |  | 支持添加已经创建的合同 |
| **title** | 合同文件标题 | Y | 50 | 每个合同在目录中体现的标题，在同一个目录下的合同标题不得重复。一般是“XXX协议”、“XXX授权书”、“XXX”  注：这里需要重新填写而不直接使用该contractId的原title，是为了防止原title加到目录后与目录中的其他合同标题冲突。 |

**示例：**

**{**

**"catalogName"**: "测试合同标题"**,**

**"contractId"**: "合同编号"**,**

**"title"**: "合同标题"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **title** | 合同文件标题 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"title"**: "XXX购车20171226"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.12、使用模版变量的多文档手动签

说明：获取合同手动签署的页面URL，将此URL发送给指定用户，即可在此页面进行手动签操作完成签署，签署位置以指定的变量位置为准。在此页面上实现“同时查看多份合同文件、只校验一次验证码，同时签署多份合同文件”的目的。查看合同时图片按合同编号排序。

请注意：本接口是签署接口，只能查看到需要签署的那几份合同，如果目录中有些合同不需要签署则不展示这些合同。

Request Path：[**server-host**]/catalog/sendByTemplate/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | | **是否必填** | **最大长度** | **默认值** |
| **catalogName** | 目录名称 | | Y |  |  |
| **signerAccount** | 指定的签署者账号 | | Y |  |  |
| **returnUrl** | 手动签署时，当用户签署完成后，签署页面指定回跳的页面地址，如果没填此参数则签署完成后跳转到合同预览页面。  如果returnurl带有“http”或“https”开头则判定h5跳转，如果returnurl不带“http”或“https”开头则判定为微信小程序跳转。 | | N |  |  |
| **vcodeMobile** | 手动签署时指定接收手机验证码的手机号，如果需要采用手动签署页面校验短信验证码，则此项必填。  如果手动签不需要短信校验，则此项可不填。 | | N |  | 20 |
| **version** | 手动签的版本，默认为2，枚举值：  2-v2版本,  3-v3版本，UI优化，移动端手绘面板自动横屏 | | N | 1 | 2 |
| **isDrawSignatureImage** | 手动签署时是否手绘签名。取枚举值0/1/2。  0点击签名图片不会触发手写面板（禁止手写）。  1点击签名图片能触发手写面板（既可手写也可使用默认签名）。  2强制必须手绘签名（只能手写不允许使用默认签名）。 | | N | 1 | 1 |
| **readAll** | 枚举值0/1，默认为0  0-无需拖动到页面底部即可确认签署  1-必须拖动到页面底部，表示阅读完毕才能确认签署 | | N |  |  |
| **signatureImageWidth** | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | | N |  |  |
| **signatureImageHeight** | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | | N |  |  |
| **isShowHandwrittenTime** | 枚举值0/1，默认为0  1-手写面板及签名图片上显示手写签名操作的时间  0-不显示该时间 | | N | 1 | 0 |
| **isFaceAuth** | 是否使用刷脸签。1-刷脸签，使用刷脸校验；  0或其他-非刷脸签，使用短信校验或无校验； | | N |  |  |
| **sid** | 流水号，开发者自定义，通常是开发者自己的业务流程中需要据此判断操作行为 | | N |  |  |
| **contractParams** | 合同组，json格式。按每个合同文件的标题分组。 | | | | |
| **参数名** | **说明** | | **最大长度** | |
| 合同文件的标题 | 每个需要签署的合同文件的标题，即“获取合同列表”接口返回的title参数的值 | |  | |
| varNames | 模板的变量名称，针对“模版”中的签署字段的变量名称，可以填多个。    变量名称获取方法有多种：  ①在开放平台编辑完模版后，将设置的变量名称记录下来；  ②用“4.21.0、获取模版变量”接口来获取，挑选出response中本次手动签需要用到的“type”为“50、40”的“name”字段值即需要使用的变量名称，拼接成字符串格式，不同变量名称之间用英文逗号间隔。  使用示例如下：  **"varNames"**: "甲方盖章1,甲方盖章2" | |  | |

**示例：**

**{**

**"catalogName"**: "per1514302565867"**,**

**"vcodeMobile"**: "13888888888"**,**

**"returnUrl"**: "http://www.baidu.com"**,**

**"contractParams"**: **{**

**"合同一的标题"**: **{**

**"varNames"**: **"stamp,stamp1,stamp2"**

**},**

**"合同二的标题"**: **{**

**"varNames"**: **"stamp,stamp1,stamp2"**

**}**

**},**

**"signerAccount"**: "per1514302565867"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

手动签署页面的URL，签署者在此页面上进行签署操作。

正常示例：

**{**

**"errno"**: 0**,**

**"cost"**: 533**,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"url"**: "https://..........."

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost"**: 533**,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

## 10、不允许WEB用户查看、下载接口合同

如果开发者要求不允许签署者通过上上签官网的注册、实名认证后查看、下载该合同，那么使用下列接口替代上述标配中的接口。

### 4.1、上传并创建合同

说明：

* 此接口是将“3.1、上传合同文件”与“4.11、创建合同”的封装在一起，方便开发者调用。调用此接口后，将文件保存到自定义的存储中，暂时只支持上传PDF文件。上传成功后得到一个合同编号，后续流程可以直接使用此合同编号进行签署。
* 如果开发者需要用到空白合同当模版，添加不同的PDF元素时，请分开使用“3.1、上传合同文件”、“3.2、PDF文件添加元素”与“4.11、创建合同”。

Request Path：**[server-host]**/storage/contract/upload/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y | 64 | 必须要指定一个用户唯一标识作为操作者 |
| **fmd5** | 文件md5值 | Y | 100 | 例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fmd5 = DigestUtils.md5Hex(bdata); |
| **ftype** | 文件类型 | Y | 10 | 仅支持PDF |
| **fname** | 文件名 | Y | 50 | 文件名必须带上后缀名，例如“XXXX.pdf” |
| **fpages** | 文件页数 | Y | 3 |  |
| **fdata** | 文件数据，base64编码 | Y | 5120000 | 例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fdata =Base64.encodeBase64String(bdata); |
| **title** | 合同标题 | Y | 50 | 客户可以将自己的业务合同编号、或合同标题放此处 |
| **expireTime** | 合同能够签署的截止时间 | Y | 20 | 合同必须在指定的截止时间之前完成签署，一旦过期则此合同将无法被签署。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”； |
| **description** | 合同描述 | N | 200 |  |
| **isAllowWebDownload** | 是否允许被上上签官网注册用户查看下载 | N | 1 | 枚举值0/1，不带此参数则默认为1。  当签署者到上上签官网注册实名认证，并登录上上签后，根据此参数控制：  0-不显示在合同列表中，无法被查看、下载  1-显示在合同列表中，可以被查看、下载 |

**示例：**

**{**

"account"**:** "mlimd@163.com"**,**

"fmd5"**:** "文件的md5值"**,**

"ftype"**:** "pdf"**,**

"fname"**:** "滴滴电子发票.pdf"**,**

"fpages"**:** "1"**,**

"fdata"**:** "文件的base64字符串"**,**

"title"**:** "测试合同标题"**,**

"description"**:** "测试合同描述"**,**

"expireTime"**:** "1531242131"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **contractId** | 创建的合同ID，在后续接口中使用 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"contractId"**: "150993706101000005"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 4.11、创建合同

说明：使用已得到的文件编号创建合同，得到合同编号，此合同还未添加签署者，属于草稿性质。每份合同必须要有创建者account。

Request Path：[**server-host**]/contract/create/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 必须要指定一个用户唯一标识作为合同的创建者（建议统一为开发者的account，或者需要进行分类统计的某些account，以便后期按此account进行统计） |
| **fid** | 文件编号 | Y |  | 上传合同文件得到的文件编号 |
| **expireTime** | 合同能够签署的截止时间 | Y |  | 合同必须在指定的截止时间之前完成签署，一旦过期则此合同将无法被签署。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”； |
| **title** | 合同标题 | Y | 50 | 客户可以将自己的业务合同编号、或合同标题放此处 |
| **description** | 合同描述 | N | 200 |  |
| **isAllowWebDownload** | 是否允许被上上签官网注册用户查看下载 | N | 1 | 枚举值0/1，不带此参数则默认为1。  当签署者到上上签官网注册实名认证，并登录上上签后，根据此参数控制：  0-不显示在合同列表中，无法被查看、下载  1-显示在合同列表中，可以被查看、下载 |

**示例：**

**{**

**"account"**: "mlimd@163.com"**,**

**"fid"**: "1702695409383636995"**,**

**"expireTime"**: "1513311132"**,**

**"title"**: "测试"**,**

**"description"**: "postman测试合同"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **contractId** | 合同编号 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"contractId"**: "1694593128888888"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 4.21.1、通过模版生成合同文件

说明：

在开放平台的“开发者模版”页面编辑保存好模版之后，就可以通过接口使用该模版。创建合同前先调本接口通过模版功能创建PDF文件。

Request Path：**[server-host]**/template/createContractPdf/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 合同创建者账号 | Y |  |  |
| **tid** | 模版编号 | Y |  | “开发者模版”页面模版保存后产生模版编号，可从页面上复制 |
| **templateValues** | 模版变量，jsonObject格式 | Y |  | 将模版变量中需要填充的内容（即变量值）在此提交。  由于在编辑模版时已经拖放了位置、并且已经保存在上上签系统中，因此这里无需再提交位置参数，只需要传入变量名称与对应的值即可。  模版变量可以在开发者模版管理页面获取。 |
| **isAllowWebDownload** | 是否允许被上上签官网注册用户查看下载 | N | 1 | 枚举值0/1，不带此参数则默认为1。  当签署者到上上签官网注册实名认证，并登录上上签后，根据此参数控制：  0-不显示在合同列表中，无法被查看、下载  1-显示在合同列表中，可以被查看、下载 |

**示例：**

**{**

**"account"**: "111222333@qq.com"**,**

**"tid"**: "1"**,**

**"templateValues"**: **{**

**"甲方姓名"**: "张三"**,**

**"甲方身份证号"**: "330101198111110311"**,**

**"甲方地址"**: "浙江省杭州市西湖区灵隐路1号"**,**

**"乙方企业名称"**: "北京朝阳群众信息技术有限公司"**,**

**"乙方证件号码"**: "330101012345678"**,**

**"乙方地址"**: "北京市朝阳区朝阳大厦1024室"

**}**

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **templateToken** | 通过模版创建合同需要这个token，格式是一个Base64编码字符串 ＋ "." + 一个自签名串，其中：  Base64编码的字符串解码后是一个json格式的字符串，里面有一些参数，其中有fid参数，对应该模版当前生成的pdf文件，如果需要，可以调用storageService进行下载。 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"templateToken "**: "一串字符串"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.2、向目录添加合同文件

说明：创建好合同目录后，将需要同时签署的合同文件添加到目录中。要一份份单独的添加，支持采用“文件编号”和“PDF文件”2种方式，2种方式二选一即可：

* 通过“文件编号”添加——实现上传好合同文件，然后在此接口使用文件编号进行添加，用到的参数为“**fid**”，当填写了fid时系统只判断fid，忽略PDF文件方式的参数；
* 通过“PDF文件”添加——在此接口提交合同PDF文件（base64编码），用到的参数为“**fmd5，fname，ftype，fpages，fdata**”，使用文件方式不需要填写fid；

Request Path：**[server-host]**/catalog/uploadContract/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **senderAccount** | 用户唯一标识 | Y | 64 | 必须要指定一个用户唯一标识作为操作者 |
| **catalogName** | 目录名称 | Y |  |  |
| **fid** | 文件编号 | Y |  | 支持添加已经上传过的文件，如果是通过fid添加，则不需要参数“**fmd5，fname，ftype，fpages，fdata**”。 |
| **title** | 合同文件标题 | Y | 50 | 每个合同文件的标题，在同一个目录下的合同标题不得重复。一般是“XXX协议”、“XXX授权书”、“XXX” |
| **fmd5** | 文件md5值 | Y | 100 | 例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fmd5 = DigestUtils.md5Hex(bdata); |
| **fname** | 文件名 | Y | 50 | 文件名必须带上后缀名，例如“XXXX.pdf” |
| **ftype** | 文件类型 | Y | 10 | 仅支持PDF |
| **fpages** | 文件页数 | Y | 3 |  |
| **fdata** | 文件数据，base64编码 | Y | 5120000 | 例如：  FileInputStream file = new FileInputStream("d: \\test\\接口系统.pdf");  byte[] bdata = IOUtils.toByteArray(file);  String fdata =Base64.encodeBase64String(bdata); |
| **expireTime** | 合同签署的到期时间 | Y | 20 | 合同必须在指定的到期时间之前完成签署，一旦过期则此合同将无法被签署。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”； |
| **description** | 合同描述 | N | 200 |  |
| **isAllowWebDownload** | 是否允许被上上签官网注册用户查看下载 | N | 1 | 枚举值0/1，不带此参数则默认为1。  当签署者到上上签官网注册实名认证，并登录上上签后，根据此参数控制：  0-不显示在合同列表中，无法被查看、下载  1-显示在合同列表中，可以被查看、下载 |

**示例：**

**{**

**"senderAccount"**: "mlimd@163.com",

**"catalogName"**: "测试合同标题",

**"fid"**: "文件编号",

**"title"**: "合同标题",

**"fmd5"**: "文件的md5值",

**"ftype"**: "pdf",

**"fname"**: "滴滴电子发票.pdf",

**"fpages"**: "1",

**"fdata"**: "文件的base64字符串",

**"description"**: "测试合同描述",

**"expireTime"**: "1531242131"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **title** | 合同文件标题 |

正常示例：

**{**

**"errno"**: 0,

**"cost": 233**,

**"data"**: **{**

**"title"**: "XXX购车20171226"

**}**,

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 8.11、向目录添加合同文件（根据contractId添加）

* 说明：创建好合同目录后，将需要同时签署的合同文件添加到目录中。已结束的目录不可再添加合同文件。要一份份单独的添加，本接口支持采用“contractId”的方式。

Request Path：[sdk-host]/catalog/addExistedContract/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **catalogName** | 目录名称 | Y |  | 已经创建的目录名称 |
| **contractId** | 合同编号 | Y |  | 支持添加已经创建的合同 |
| **title** | 合同文件标题 | Y | 50 | 每个合同在目录中体现的标题，在同一个目录下的合同标题不得重复。一般是“XXX协议”、“XXX授权书”、“XXX”  注：这里需要重新填写而不直接使用该contractId的原title，是为了防止原title加到目录后与目录中的其他合同标题冲突。 |
| **isAllowWebDownload** | 是否允许被上上签官网注册用户查看下载 | N | 1 | 枚举值0/1，不带此参数则默认为1。  当签署者到上上签官网注册实名认证，并登录上上签后，根据此参数控制：  0-不显示在合同列表中，无法被查看、下载  1-显示在合同列表中，可以被查看、下载 |

**示例：**

**{**

**"catalogName"**: "测试合同标题"**,**

**"contractId"**: "合同编号"**,**

**"title"**: "合同标题"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **title** | 合同文件标题 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"title"**: "XXX购车20171226"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

## 国密证书服务

本服务提供的是使用国密算法（SM2）完成“申请证书、签署”的动作，本服务的接口用于替代普通接口。

### 11.1、申请国密数字证书

说明：

1. 为用户申请国密算法数字证书，该数字证书用于电子签名，本接口用于替代接口“1.1、注册个人用户并申请证书”、“1.2、注册企业用户并申请证书”、“1.9、同步申请数字证书”。
2. 使用本接口申请数字证书前必须按此流程使用：先调用“1.6、注册用户”，然后调用“1.7、设置个人用户证件信息”或“1.8、设置企业用户证件信息”接口。
3. 用户必须先申请数字证书，否则将无法完成签署。

Request Path：[**server-host**]/user/applySM2Cert/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 为哪个用户申请证书就填该用户的唯一标识 |

**示例：**

**{**

**"account"**: "mlimd@163.com"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **certId** | 证书编号 |
| **certType** | 证书类型 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"certType"**: "CFCA"**,**

**"certId"**: "CFCA#SM2-33-20170217170217942-65183"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 11.2、查询国密证书编号

说明：查询某一用户的国密数字证书编号，如果该用户申请过国密数字证书则返回，本接口用于替代“1.3、查询证书编号”。

Request Path：[**server-host**]/user/getSM2Cert/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **account** | 用户唯一标识 | Y |  | 查询哪个用户的证书就填该用户的唯一标识 |

**示例：**

**{**

**"account"**: "1674224491900698632"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **account** | 用户唯一标识 |
| **certId** | 证书编号 |

正常示例：

**{**

**"errno"**: 0**,**

**"errmsg"**: ""**,**

**"cost": 233,**

**"data"**: **{**

**"account"**: "1674224491900698632"**,**

**"certId"**: "CFCA#SM2-33-20171010142326279-12345"

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"data"**: null**,**

**"cost": 233,**

**"errmsg"**: "check sign wrong "

**}**

### 11.3、用国密证书签署合同（即自动签）

说明：使用国密证书签署合同，可直接用于无感知的快捷签署。此接口要求签署者必须拥有自己的国密数字证书。可以无需先调用“添加签署者”接口与“设置合同签署参数”接口。本接口用于替代“4.2、签署合同（即自动签）”。

Request Path：**[server-host]**/contract/sign/certSM2/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | |
| **contractId** | 合同编号 | Y |  |  | | |
| **signer** | 签署者 | Y |  | 即签署者的account | | |
| **signaturePositions** | 指定的签署位置，json array格式 | Y |  | 每一项为一个json对象 | | |
|  |  | 最大长度 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| y | 纵坐标，同上 | 10 |
| pageNum | 页码。1开始，不能超过实际页数 | 3 |
| rptPageNums | 当前位置的签名需要复制到的目标页码列表。该参数用于控制是否将当前位置的签名复制到其他页。参数启用之后，当前页的签名会同时出现在其他页的相同位置。参数值格式如下： 多个页码使用英文逗号(,)分隔。 参数不存在或者参数值为空表示不复制， 参数值为0表示复制到所有页， 参数值为其他表示复制到指定页。 参数值示例："0" "1,2,3,4,5,6" "1,3,5,7,9" |  |
| **signatureImageName** | 签名图片名称 | N |  | 企业用户可以指定上传过的印章图片中的某一张作为本次签署的印章图片，取“上传用户签名/印章图片”接口设置的**imageName**的值 | | |
| **signatureImageData** | 签名图片 | N |  | 用户指定的签名图片，Base64字符串。允许为空，为空时使用用户默认的签名图片。  使用自动签的用户，可以越过“2.1、创建用户签名/印章图片”接口、或“2.2、上传用户签名/印章图片”接口，直接在此提供签名/印章图片即可。 | | |
| **signatureImageWidth** | 签名图片显示宽度 | N |  | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。 如果有一个为空，则按照同时为空处理，即使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。 如果同时有值，则图片按照两个宽高值拉伸填充，效果可能会产生变形。 | | |
| **signatureImageHeight** | 签名图片显示高度 | N |  | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。 如果有一个为空，则按照同时为空处理，即使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。 如果同时有值，则图片按照两个宽高值拉伸填充，效果可能会产生变形。 | | |

**示例：**

**{**

**"contractId"**: "cU34344234"**,**

**"signer"**: "u-1112343242"**,**

**"signatureImageData"**: "cU34344234"**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.05"**,**

**"y"**: "0.71"**,**

**"rptPageNums"**: "2,3,4,5,6,7"

**},**

**{**

**"pageNum"**: "2"**,**

**"x"**: "0.05"**,**

**"y"**: "0.81"

**}**

**]**

**}**

**响应（RESPONSE）：**

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{ },**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 11.4、用模版变量和国密证书签署合同

说明：

使用模版功能创建合同后，调用本接口使用国密证书完成签署，本接口用于替代“4.21.3、用模版变量签署合同”。

Request Path：**[server-host]**/contract/sign/templateSM2/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  |
| **contractId** | 合同编号 | Y |  |  |
| **tid** | 模版编号 | Y |  |  |
| **vars** | 模版变量值 | Y | JSONObject格式 | 每一项为一个jsonObject，格式类似如下：  **varName——**模版变量名，比如“甲方盖章”、“甲方签署日期”等   |  |  | | --- | --- | | **参数** | **说明** | | **account** | 签署者账号，同一份合同上的同一个签署者不可分多次签署。不同的签署者分次签署。 | | **signatureImageData** | 签名图片，base64编码字符串。   * 如果事先调用上传签名图片接口为签署者上传过签名图片，则此处可以不传，如果传了则优先使用此处上传的签名图片。 * 如果变量是签署日期类型则此项不传，传了会报错。 | |
| **loginIp** | 登录IP | N |  | 签署者登录开发者平台的IP地址 |
| **loginTime** | 登录时间 | N |  | 签署者登录开发者平台的登录时间 |
| **signIp** | 签署IP | N |  | 签署者操作通过开发者平台操作签署时的IP |

**示例：**

**{**

**"contractId"**: ""**,**

**"vars"**: **{**

**"甲方盖章"**: **{**

**"account"**: ""**,**

**"signatureImageData"**: "base64编码"

**},**

**"甲方签署日期"**: **{**

**"account"**: ""

**}**

**},**

**"tid"**: ""**,**

**"signIp"**: ""**,**

**"loginIp"**: ""**,**

**"loginTime"**: ""

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：无

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{** **},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 11.5、注册个人用户并申请国密证书

说明：创建一个上上签个人用户，并提交用户的证件信息，为该用户申请国密数字证书。后续有关该用户的所有操作都使用该用户的account。本接口用于替代接口“1.1、注册个人用户并申请证书”。

由于证书是异步申请，不会立即返回证书申请结果，所以在签署之前，需要确认一次该用户的证书是否申请成功，调用接口“1.11、异步申请状态查询”确认。

Request Path：[**server**-host]/user/regWithApplySM2Cert/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | | | |
| **account** | 用户帐号 | Y |  | 用户的唯一标识，可以是邮箱、手机号、证件号等不限 | | | | |
| **name** | 用户名称 | Y | 100 | 必须和证件上登记的姓名一致 | | | | |
| **userType** | 用户类型 | Y | 1 | 1表示个人 | | | | |
| **mail** | 用户邮箱 | N | 50 |  | | | | |
| **mobile** | 用户手机号 | N | 20 |  | | | | |
| **credential** | 用户证件信息对象 | Y |  | 用户证件信息对象，json格式么具体参数如下： | | | | |
| **参数** | **说明** | **是否必填** | **最大长度** |  |
| **identity** | 用户证件号 | Y | 30 | 必须和证件上登记的号码一致 |
| **identityType** | 用户证件类型 | Y | 2 | 默认为“0”, 0-居民身份证  1-护照  B-港澳居民往来内地通行证  C-台湾居民来往大陆通行证  E-户口簿  F-临时居民身份证 |
| **contactMobile** | 联系手机 | N | 20 |  |
| **contactMail** | 联系邮箱 | N | 50 |  |
| **province** | 省份 | N | 10 |  |
| **city** | 城市 | N | 50 |  |
| **address** | 地址 | N | 100 |  |

**示例：**

**{**

**"account"**: "test@bestsign.cn"**,**

**"mail"**: "test@bestsign.cn"**,**

**"mobile"**: "13800001234"**,**

**"name"**: "张三"**,**

**"userType"**: "2"**,**

**"credential"**: **{**

**"identity"**: "100123199001010011"**,**

**"identityType"**: "0"**,**

**"contactMail"**: "test@bestsign.cn"**,**

**"contactMobile"**: "13800001234"**,**

**"province"**: "浙江省"**,**

**"city"**: "杭州市"**,**

**"address"**: "xx路xx号"

**},**

**"applyCert"**: "1"

**}**

**响应（RESPONSE）：**

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **taskId** | 异步申请证书队列中的任务编号，在24小时内可用于查询异步申请状态，**taskId**过24小时后就失效 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"taskId"**: "170217170217942"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 11.6、注册企业用户并申请国密证书

说明：创建一个上上签企业用户，并提交用户的证件信息，为该用户申请国密数字证书。后续有关该用户的所有操作都使用该用户的account。本接口用于替代接口“1.2、注册企业用户并申请证书”。

由于证书是异步申请，不会立即返回证书申请结果，所以在签署之前，需要确认一次该用户的证书是否申请成功，调用接口“1.11、异步申请状态查询”确认。

Request Path：[**server**-host]/user/regWithApplySM2Cert/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** |  | | | | |
| **account** | 用户帐号 | Y |  | 用户的唯一标识，可以是邮箱、手机号、证件号等不限 | | | | |
| **name** | 企业名称 | Y | 100 | 必须和企业证件上登记的名称一致  如个体工商户在营业执照上无企业名称的，请填“经营者”名字 | | | | |
| **userType** | 用户类型 | Y | 1 | 2表示企业 | | | | |
| **mail** | 用户邮箱 | N | 50 |  | | | | |
| **mobile** | 用户手机号 | N | 20 |  | | | | |
| **credential** | 企业证件信息对象 | Y |  | 企业证件信息对象，json，  三证合一的统一社会信用代码这样传：regcode = "统一社会信用代码"orgcode = "统一社会信用代码"taxcode = "统一社会信用代码"老三证这样传：regcode = "工商注册号"orgcode = "组织机构代码证"taxcode = "税务登记证"个体户这样传：regcode = "工商注册号"orgcode = ""taxcode = ""  具体参数如下： | | | | |
| **参数** | **说明** | **是否必填** | **最大长度** |  |
| **regCode** | 工商注册号 | Y | 30 | 企业用户设置. 如果是三证合一，regCode/taxCode/orgCode三个填一样的值  regCode和orgCode必须输入一个，否则异常返回：240006: regCode/orgCode must be have one |
| **orgCode** | 组织机构代码 | Y | 30 | 企业用户设置. 如果是三证合一，regCode/taxCode/orgCode三个填一样的值  regCode和orgCode必须输入一个，否则异常返回：240006: regCode/orgCode must be have one |
| **taxCode** | 税务登记证号 | Y | 30 | 企业用户设置. 如果是三证合一，regCode/taxCode/orgCode三个填一样的值 |
| **legalPerson** | 法定代表人姓名 | Y | 50 | 法定代表人姓名或经办人姓名 |
| **legalPersonIdentity** | 法定代表人证件号 | Y | 30 | 法定代表人证件号或经办人证件号 |
| **legalPersonIdentityType** | 法定代表人证件类型 | Y | 2 | 法定代表人证件类型或经办人证件类型，与“**legalPersonIdentity**”要匹配，默认为“0”。  0-居民身份证  1-护照  B-港澳居民往来内地通行证  C-台湾居民来往大陆通行证  E-户口簿  F-临时居民身份证 |
| **legalPersonMobile** | 法定代表人手机号 | N | 20 | 法定代表人手机号或经办人手机号 |
| **contactMobile** | 联系手机 | Y | 20 | 联系手机必填，为CA年检抽查时联系使用 |
| **contactMail** | 联系邮箱 | N | 50 | **credential对象属性** |
| **province** | 省份 | N | 10 | **credential对象属性** |
| **city** | 城市 | N | 50 | **credential对象属性** |
| **address** | 地址 | N | 100 | **credential对象属性** |

**示例：**

**{**

**"account"**: "test@bestsign.cn"**,**

**"mail"**: "test@bestsign.cn"**,**

**"mobile"**: "13800001234"**,**

**"name"**: "浙江xx公司"**,**

**"userType"**: "2"**,**

**"credential"**: **{**

**"regCode"**: "5566789"**,**

**"taxCode"**: "5566789"**,**

**"orgCode"**: "5566789"**,**

**"contactMail"**: "test@bestsign.cn"**,**

**"contactMobile"**: "13800001234"**,**

**"legalPerson"**: "张三"**,**

**"legalPersonIdentity"**: "100123199001010011"**,**

**"legalPersonIdentityType"**: "0"**,**

**"legalPersonMobile"**: "17712341234"**,**

**"province"**: "浙江省"**,**

**"city"**: "杭州市"**,**

**"address"**: "xx路xx号"

**},**

**"applyCert"**: "1"

**}**

**响应（RESPONSE）：**

|  |  |
| --- | --- |
| **参数名** | **说明** |
| **taskId** | 异步申请证书队列中的任务编号，在24小时内可用于查询异步申请状态，**taskId**过24小时后就失效 |

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"data"**: **{**

**"taskId"**: "170217170217942"

**},**

**"errmsg"**: ""

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 11.8、用国密证书的手动签

说明：获取合同手动签署的页面URL，将此URL发送给指定用户，即可在此页面进行手动签操作完成签署。增加了参数“**signatureImageWidth**”与“**signatureImageHeight**”，可以指定签名图片显示的宽高。本接口使用了国密证书，用来替代标准流程中的RSA算法的手动签。

Request Path：[**server-host**]/contract/sendWithSM2/

Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **参数名** | **说明** | | | **是否必填** | **最大长度** | **默认值** |
| **contractId** | 合同编号 | | | Y |  |  |
| **signer** | 指定给哪个用户看 | | | Y |  |  |
| **dpi** | 预览图片清晰度，枚举值：96-低清（默认），120-普清，160-高清，240-超清 | | | N | 3 | 96 |
| **signaturePositions** | **参数名** | **说明** | **最大长度** | Y |  |  |
| y | 纵坐标，同上 | 10 |
| x | 横坐标，按页面尺寸的百分比计算，取值0.0 - 1.0。以左上角为原点 | 10 |
| pageNum | 页码，从1开始，不能超过合同最大页数 | 3 |
| rptPageNums | 当前位置的签名需要复制到的目标页码列表。该参数用于控制是否将当前位置的签名复制到其他页。参数启用之后，当前页的签名会同时出现在其他页的相同位置。参数值格式如下： 多个页码使用英文逗号(,)分隔。 参数不存在或者参数值为空表示不复制， 参数值为0表示复制到所有页， 参数值为其他表示复制到指定页。 参数值示例："0" "1,2,3,4,5,6" "1,3,5,7,9" |  |
| type | 日期类型专用，当签署位置是“日期”类型的签名时，本参数需要填写“date” |  |
| dateTimeFormat | 日期类型专用，当签署位置是“日期”类型的签名时，本参数需要填写日期的格式，格式支持以下类型：  MM-dd-yyyy  yyyy-MM-dd  MM.dd.yyyy  Yyyy.MM.dd  MM/dd/yyyy  Yyyy/MM/dd |  |
| fontSize | 日期类型专用，当签署位置是“日期”类型的签名时，本参数需要填写字号大小 |  |
| **isAllowChangeSignaturePosition** | 在有指定signaturePOSTion参数的情况下，是否允许拖动签名位置。取值1/0。（0：不允许，1：允许） | | | N | 1 | 0 |
| **expireTime** | 签署链接的到期时间，unix时间戳格式（秒数）。超过此时间则无法打开链接页面，需要获取新的签署合同链接。格式为秒级的unix时间戳，如希望“2017/12/30 10:21:52”到期，则设置为“1514600512”。  注：并不是合同的可签署到期时间，只是此签署链接的有效期。  由于返回的是短链接，如果没有设置则默认是7天，如果设置了以设置的为准，但有效期最大不能超过7天。 | | | N | 20 | 0 |
| **returnUrl** | 手动签署时，当用户签署完成后，签署页面指定回跳的页面地址，如果没填此参数则签署完成后跳转到合同预览页面。  如果returnurl带有“http”或“https”开头则判定h5跳转，如果returnurl不带“http”或“https”开头则判定为微信小程序跳转。 | | | N |  |  |
| **vcodeMobile** | 手动签署时指定接收手机验证码的手机号，如果需要采用手动签署页面校验短信验证码，则此项必填。  如果手动签不需要短信校验，则此项可不填。  自动签可不填此项。 | | | N |  | 20 |
| **isDrawSignatureImage** | 手动签署时是否手绘签名。取值0/1/2。  0点击签名图片不会触发手写面板（禁止手写）。  1点击签名图片能触发手写面板（既可手写也可使用默认签名）。  2强制必须手绘签名（只能手写不允许使用默认签名），初始不显示用户签名图片，只显示实时手写的签名。 | | | N | 1 | 1 |
| **signatureImageName** | 签名/印章图片。取值default或者指定的印章名称。（手动签只有参数version=3时才支持企业多印章，填写自定义的印章名称） | | | N | default | 50 |
| **sid** | 开发者自定义的签署流水号（比如第几次签署之类），如有填写，则在异步推送签署结果时包含此签署流水号 | | | N | 30 |  |
| **pushUrl** | 此处有配置则签署推送消息优先使用此配置，如果此处没有配置，则取开发者统一配置的异步推送地址 | | | N |  |  |
| **version** | 手动签的版本，默认为2，枚举值：  2-v2版本,  3-v3版本，UI优化，移动端手绘面板自动横屏 | | | N | 1 | 2 |
| **readAll** | 枚举值0/1，默认为0  0-无需拖动到页面底部即可确认签署  1-必须拖动到页面底部，表示阅读完毕才能确认签署 | | | N |  |  |
| **app** | app 人脸比对签署（即app内刷脸签署）用到的app参数，参数内容为JSBridge，即手动签页面将此参数传递给客户app以便客户app调用对应的SDK。 一般情况，如果不需要用户跳出手动签页面执行其他app操作的话，则REQUEST BODY不需要带有此参数。  新版手动签在返回签署人配置的时候，会将此参数返回给前端，前端需要将此参数传递给app。 | | | N |  |  |
| **signatureImageWidth** | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | | | N |  |  |
| **signatureImageHeight** | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | | | N |  |  |
| **isShowHandwrittenTime** | 枚举值0/1，默认为0   1. 手写面板及签名图片上显示手写签名操作的时间   0-不显示该时间 | | | N | 1 | 0 |
| **isFaceAuth** | 是否使用刷脸签。   1. 刷脸签，使用刷脸校验；   0或其他-非刷脸签，使用短信校验或无校验； | | | N |  |  |

**示例：**

**{**

**"contractId"**: "cU34344234"**,**

**"expireTime"**: "0"**,**

**"dpi"**: "120"**,**

**"signer"**: ""**,**

**"signaturePositions"**: **[**

**{**

**"pageNum"**: "1"**,**

**"x"**: "0.4622"**,**

**"y"**: "0.5852999973297119"**,**

**"rptPageNums"**: ""**,**

**"type"**: "date"**,**

**"dateTimeFormat"**: "MM-dd-yyyy"**,**

**"fontSize"**: "18"

**},**

**{**

**"pageNum"**: "6"**,**

**"x"**: "0.4622"**,**

**"y"**: "0.7852999973297119"

**}**

**],**

**"isAllowChangeSignaturePosition"**: "0"**,**

**"returnUrl"**: ""**,**

**"vcodeMobile"**: ""**,**

**"isDrawSignatureImage"**: "1"**,**

**"signatureImageName"**: "default"**,**

**"pushUrl"**: "https://xxxxxxx"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

手动签署页面的url，签署者在此页面上进行签署操作。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"url"**: "https://..........."

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

### 11.9、用国密证书的多文档手动签

说明：获取合同手动签署的页面URL，将此URL发送给指定用户，即可在此页面进行手动签操作完成签署。在此页面上实现“同时查看多份合同文件、只校验一次验证码，同时签署多份合同文件”的目的。查看合同时图片按合同编号排序。本接口使用了国密证书，用来替代标准流程中的RSA算法的手动签。

请注意：本接口是签署接口，只能查看到需要签署的那几份合同，如果目录中有些合同不需要签署则不展示这些合同。

Request Path：[**server-host**]/catalog/send/

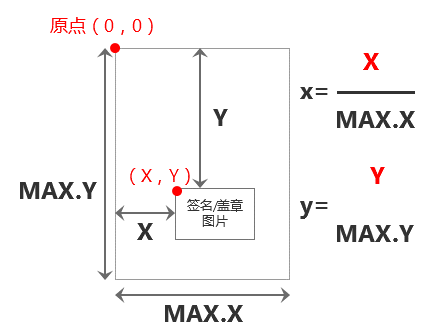
Method：POST

其他url参数：无

需要参与签名的其他URL参数：无

**请求参数（REQUEST BODY）：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数名** | **说明** | **是否必填** | **最大长度** | **默认值** |
| **catalogName** | 目录名称 | Y |  |  |
| **signerAccount** | 指定的签署者账号 | Y |  |  |
| **returnUrl** | 手动签署时，当用户签署完成后，签署页面指定回跳的页面地址，如果没填此参数则签署完成后跳转到合同预览页面。  如果returnurl带有“http”或“https”开头则判定h5跳转，如果returnurl不带“http”或“https”开头则判定为微信小程序跳转。 | N |  |  |
| **vcodeMobile** | 手动签署时指定接收手机验证码的手机号，如果需要采用手动签署页面校验短信验证码，则此项必填。  如果手动签不需要短信校验，则此项可不填。 | N |  | 20 |
| **dpi** | 预览图片清晰度，枚举值：96-低清（默认），120-普清，160-高清，240-超清 | N | 3 | 96 |
| **version** | 手动签的版本，默认为2，枚举值：  2-v2版本,  3-v3版本，UI优化，移动端手绘面板自动横屏 | N | 1 | 2 |
| **isDrawSignatureImage** | 手动签署时是否手绘签名。取枚举值0/1/2。  0点击签名图片不会触发手写面板（禁止手写）。  1点击签名图片能触发手写面板（既可手写也可使用默认签名）。  2强制必须手绘签名（只能手写不允许使用默认签名）。 | N | 1 | 1 |
| **readAll** | 枚举值0/1，默认为0  0-无需拖动到页面底部即可确认签署  1-必须拖动到页面底部，表示阅读完毕才能确认签署 | N |  |  |
| **signatureImageWidth** | 本次签署使用的签名/印章图片在合同PDF上显示的宽度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | N |  |  |
| **signatureImageHeight** | 本次签署使用的签名/印章图片在合同PDF上显示的高度，72dpi下的磅值。如果指定位置上没有设置签名图片的宽度和高度，那么按照如下默认值处理： 个人签名：默认高度：48磅，最大宽度：130磅，宽度一般小于等于130磅。 企业印章：默认宽度：130磅，默认高度：130磅。  注意：signatureImageWidth和signatureImageHeight必须同时为空，或者同时存在。  同时为空则使用系统默认的高度和宽度（宽度和高度最大130磅，小于130磅的使用原始尺寸，大于130磅的按比例缩放）。  如果同时有值，则图片按照两个宽高值拉伸/压缩填充，效果可能会产生变形。 | N |  |  |
| **isShowHandwrittenTime** | 枚举值0/1，默认为0  1-手写面板及签名图片上显示手写签名操作的时间  0-不显示该时间 | N | 1 | 0 |
| **isFaceAuth** | 是否使用刷脸签。1-刷脸签，使用刷脸校验；  0或其他-非刷脸签，使用短信校验或无校验； | N |  |  |
| **sid** | 流水号，开发者自定义，通常是开发者自己的业务流程中需要据此判断操作行为 | N |  |  |
| **contractParams** | 合同组，json格式。按每个合同文件的标题分组。 | | | |
| **参数名** | **说明** | **最大长度** | |
| 合同文件的标题 | 每个需要签署的合同文件的标题，即“获取合同列表”接口返回的title参数的值 |  | |
| signaturePositions | json数组格式，包含3个变量：pageNum、x、y |  | |
| pageNum | 页码，从1开始，不能超过合同最大页数 | 3 | |
| x | 横坐标，按页面尺寸的百分比计算，取值大于0小于1的小数，允许小数点。以左上角为原点 | 10 | |
| y | 纵坐标，同上 | 10 | |



**坐标像素换算公式：**

文档页面的左上角为坐标原点（0，0）

X：签名图片左上角离原点的实际X轴距离

Y：签名图片左上角离原点的实际Y轴距离

MAX\_x：文档页面的最大宽度

MAX\_y：文档页面的最大高度

X/MAX\_x=x表示换算后的x坐标比例x

Y/MAX\_y=y表示换算后的y坐标比例y

**示例：**

**{**

**"catalogName"**: "per1514302565867"**,**

**"vcodeMobile"**: "13888888888"**,**

**"returnUrl"**: "http://www.baidu.com"**,**

**"contractParams"**: **{**

**"合同一的标题"**: **{**

**"signaturePositions"**: **[**

**{**

**"x"**: "0.5"**,**

**"y"**: "0.5"**,**

**"pageNum"**: 1

**},**

**{**

**"x"**: "0.5"**,**

**"y"**: "0.5"**,**

**"pageNum"**: 2

**}**

**]**

**},**

**"合同二的标题"**: **{**

**"signaturePositions"**: **[**

**{**

**"x"**: "0.5"**,**

**"y"**: "0.5"**,**

**"pageNum"**: 1

**},**

**{**

**"x"**: "0.5"**,**

**"y"**: "0.5"**,**

**"pageNum"**: 2

**}**

**]**

**}**

**},**

**"signerAccount"**: "per1514302565867"

**}**

**响应（RESPONSE）：**

RESPONSE DATA参数如下：

手动签署页面的URL，签署者在此页面上进行签署操作。

正常示例：

**{**

**"errno"**: 0**,**

**"cost": 233,**

**"errmsg"**: ""**,**

**"data"**: **{**

**"url"**: "https://..........."

**}**

**}**

异常示例：

**{**

**"errno"**: 240012**,**

**"cost": 233,**

**"data"**: null**,**

**"errmsg"**: "check sign wrong "

**}**

# 六、附录：返回信息编号列表

通用错误码：

|  |  |
| --- | --- |
| **errno** | **说明** |
| 0 | 成功 |
| 249998 | unknown error 系统未知错误 |
| 249999 | service pause 系统暂停 |
| 240001 | 系统产生的错误 |
| 240002 | 系统超时 |
| 240003 | system busy, please try later 系统忙 |
| 240004 | 配置异常 |
| 240005 | 执行异常 |
| 240006 | 参数传递异常 |
| 240007 | 不存在 异常 |
| 240008 | 已存在 异常 |
| 240009 | 系统禁止异常 |
| 240010 | 不支持 |
| 240011 | 过期异常 |
| 240012 | 验证签名错误异常 |
| 240013 | 加锁异常 |
| 240014 | 解锁异常 |
| 240015 | 发送SMS信息一场 |
| 240016 | 验证码错误 |
| 240017 | 无效的URL |
| 与开发者相关 |  |
| 241107 | 开发者不存在 |
| 241108 | 开发者已经存在 |
| 241109 | 开发者被禁用 |
| 241112 | 开发者签名错误 |
| 241115 | 开发者没有认证的key |
| 241116 | 开发者没有RSA key |
| 241117 | 开发者没有相应的配置项 |
| 与用户相关 |  |
| 241207 | 用户不存在 |
| 241208 | 用户已经存在 |
| 241209 | 用户被禁止 |
| 241217 | 用户签名图片不存在 |
| 241218 | 用户签名已经存在 |
| 241227 | 用于认证的信息不存在 |
| 241228 | 用户注册失败 |
| 241229 | 保存认证信息失败 |
| 241230 | 上传签名图片失败 |
| 与证书相关 |  |
| 241303 | 申请证书超时 |
| 241307 | 证书不存在 |
| 241308 | 证书已经存在 |
| 241309 | 证书被禁用 |
| 241310 | 证书不匹配 |
| 241317 | 证书申请失败 |
| 与合同相关 |  |
| 241407 | 合同不存在 |
| 241410 | 不是合同的拥有者 |
| 241411 | 合同未结束 |
| 241412 | 合同签署错误 |
| 241413 | 合同不能被拒绝 |
| 241414 | 合同不能结束 |
| 241415 | 合同不能被操作 |
| 241416 | 合同结束错误 |
| 241417 | 更新合同状态错误 |
| 241418 | 创建合同失败 |
| 241419 | 添加签署人失败 |
| 241420 | 合同签署失败 |
| 241421 | 合同没有签署人 |
| 241422 | 手动签错误 |
| 241423 | 合同已结束不能被操作 |
| 241424 | signer signed or canceled（签署者已签署或合同已被取消） |
| 241426 |  |
| 与微服务相关 |  |
| 241501 | 微服务调用异常 |

手动签有关错误码：

|  |  |  |
| --- | --- | --- |
| **errno** | **英文** | **说明** |
| 240006 | no developerId | 没有developerId参数 |
| 240006 | bad developerId | developerId不是正确的格式 |
| 240006 | no contractId | 没有contractId参数 |
| 240006 | bad contractId | contractId不是正确格式 |
| 240006 | no account | 没有account |
| 240006 | no sendType | 没有sendType |
| 240006 | bad seconds | seconds参数格式不对 |
| 240006 | no vcodeMobile set of contractConfig | 合同配置项中没有发送验证码的mobile |
| 240006 | bad vcodeMobile set of contractConfig | mobile格式错误 |
| 240006 | no Verification code | 没有验证码 |
| 240006 | signaturePositions's page number out of bounds | 签署位置设置越界 |
| 240007 | contract not exists | 合同不存在 |
| 240007 | fileMeta not exists | 文件结构信息不存在 |
| 241207 | user not exists | 用户不存在 |
| 241421 | not signer | 不是签署者 |
| 240016 | only send a verification code in 60s for a phone | 60秒内只允许一台手机只允许发送一次验证码 |
| 240016 | Verification code wrong | 验证码验证错误 |
| 240016 | failed to verify vcode | 验证码校验失败 |
| 240020 | vcode not exists | 验证码过期或不存在 |

# 文档修订记录

|  |  |  |
| --- | --- | --- |
| 修订日期 | 内容 | 修订人 |
| 2018.10.30 | 1. 接口“3.2、转换文件格式”从标配挪到选配接口列表； 2. 新增接口“4.21.6、获取创建模版的地址”； 3. 新增接口“4.21.7、获取编辑模版的地址”； 4. 新增接口“4.21.8、获取开发者模版列表”； | 杜晓杰 |
| 2018.10.31 | 1. 新增选配接口“8.12、使用模版变量的多文档手动签”； 2. 将“4.28、合同批量签署（即批量自动签）”、“4.29、合同关键字批量签署（即批量自动签）”移动到选配列表； 3. 新增选配接口“4.37、合同PDF文件上添加附件”； | 杜晓杰 |
| 2018.11.06 | 1、接口“4.37、合同PDF文件上添加附件”的Request Path改为“/dist/contract/pdfAddAttachment/” | 杜晓杰 |
| 2018.11.15 | 1. 接口“4.21.6、获取创建模版的地址”增加参数sid，并添加异步通知格式描述； 2. 接口“4.21.7、获取编辑模版的地址”增加参数sid，并添加异步通知格式描述； 3. 接口“8.4、获取多文档手动签署URL”增加参数sid； 4. 接口“8.12、使用模版变量的多文档手动签”增加参数sid； | 杜晓杰 |
| 2018.11.22 | 接口“8.4、获取多文档手动签署URL”增加参数dpi | 杜晓杰 |
| 2018.12.11 | 1、单文档手动签、模版手动签添加“签署日期”的签名类型；  2、新增支付服务的2个接口：“12.1、微信支付”接口、“12.2查询支付订单”接口  3、异步通知增加支付接口的异步通知；  4、新增“4.21.9、预览模版”接口 | 杜晓杰 |
| 2018.12.28 | 新增“11.8、用国密证书的手动签”、“11.9、用国密证书的多文档手动签” | 杜晓杰 |
| 2019.01.24 | 4.28发送合同（即手动签，指定图片大小）添加签署方式参数 | 祖忠慧 |