CEC

2.5.0.0.0

User Access--Web Client Access (RESTful)

Issue	01
Date	2024-03-01

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: https://www.huawei.com

Email: support@huawei.com

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process*. For details about this process, visit the following web page:
https://www.huawei.com/en/psirt/vul-response-process
For vulnerability information, enterprise customers can visit the following web page:
https://securitybulletin.huawei.com/enterprise/en/security-advisory

Contents

1 Overview

The CEC provides RESTful APIs for third parties to develop agent clients. These APIs need to be exposed on the public network by the API Fabric for third parties to use. The following sections describe the open APIs.

1.1 RESTful API Overview

RESTful APIs are used to create, read, update, and delete (CRUD) resources and data using HTTP/HTTPS and URIs.

Third-party systems can use HTTP/HTTPS GET and POST methods to perform operations on system resources and data.

NOTE

HTTP is not recommended because it may bring risks. The secure HTTPS is recommended.

RESTful APIs can be invoked only in the same trusted zone. If the interfaces need to be opened to the Internet, the OpenAPI GW/AEP needs to be introduced as the authentication platform.

1.2 RESTful API Format

This section describes the format of request and response messages of RESTful APIs.

1.2.1 Request Message

A request message consists of the request line, request header, and request body (optional). **Table 1-1** describes the request message format.

Table 1-1 Request message format

HTTP/HTTPS Request Message	
HTTP/ HTTPS request line	Method Request-URI HTTP-Version **NOTE** There is a space between Method and Request-URI, and between Request-URI and HTTP-Version. There is a carriage return behind HTTP-Version.
Request header	Content-Type:application/json;charset=UTF-8
Blank line	CRLF
Message body (optional)	Business request message (JSON description)

Request Line

A request line starts with a method token, followed by the request URI and the protocol version. There is a space between Method and Request-URI, and between Request-URI and HTTP-Version. The request line format is as follows:

Method Request-URI HTTP-Version CRLF

- Method: request method, for example, GET, POST, PUT, and DELETE.
- Request-URI: uniform resource identifier. For example, the universal URI format is https://10.10.10.2:9443/balancemgmt/v1/balance.
- HTTP-Version: HTTP protocol version of the request, for example, HTTP/1.1.
- CRLF: carriage return and line feed character. CRLF is placed only at the end of a line. CR and LF must be present at the same time.

Request Header

A request header consists of multiple header fields. Each header field consists of a field name and value, which are separated by a colon (:). A request header at the virtualization layer generally includes the following header fields:

- Host: https://<hostname>[:<port>]

  The default port number for HTTPS is 9443.
- Content-Type: application/json

  Indicates that the message body type is JSON.
- Accept: application/json;version=1.0
- Accept-Language:zh_CN

  Specifies the language of the response. The options are zh_CN (simplified Chinese) and en_US (English).
- Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==

  Authentication field, which uses a user ID and password for authentication.

- AccessKeyId: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

  Access ID of an account or a user, which identifies the owner who creates the resource. This field is mandatory when the C3 business interface is invoked and optional when the C3 maintenance interface is invoked.

Request Body

The request body is in JSON format. For details, see the definition of each API in the following sections.

1.2.2 Response Message

A response message consists of a status line, response header, and response body. **Table 1-2** describes the response message format.

Table 1-2 Response message format

HTTP/HTTPS Response Message	
HTTP/ HTTPS response line	HTTP-Version Status-Code Reason-Phrase **NOTE** There is a space between HTTP-Version and Status-Code, and between Status-Code and Reason-Phrase. There is a carriage return after Reason-Phrase.
Response header	Content-Type:application/json;charset=UTF-8
	...
Blank line	CRLF
Message body (optional)	Business request message (JSON description)

Status Line

HTTP-Version Status-Code Reason-Phrase CRLF

In the preceding information:

- HTTP-Version: HTTP version number of the server.
- Status-Code: response status code returned by the server.
- Reason-Phrase: text description of a status code.

  A status code consists of three digits. The first digit defines the response type and has five options.

  – 1xx: Informational - The request has been received and can be further processed.

  – 2xx: Success - The request has been successfully received, parsed, and accepted.

  – 3xx: Redirection - Further operations must be performed for completing the request.

  - 4xx: Client error - The request has syntax errors or cannot be implemented.
  - 5xx: Server error - The server fails to fulfill a valid request.

  Common status codes are as follows:

  - 200 OK: The request has succeeded.
  - 400 Bad Request: The client request has syntax errors and cannot be parsed by the server.
  - 401 Unauthorized: The request is unauthorized.
  - 403 Forbidden: The server receives the request but refuses to provide services.
  - 404 Not Found: The requested resource does not exist.
  - 500 Internal Host Error: An unexpected error occurs on the server.
  - 503 Host Unavailable: The server cannot process the client request now and may be able to process the request later.

  Most of the time, 200 OK is returned:

  HTTP/1.1 200 OK (CRLF)

Response Header

A response header usually contains the following header fields:

- Date: Mon, 12 Nov 2007 15:55:01 GMT

  Indicates the response sending time.

- Content-Length: xxx

  Specifies the length of the entity text and is stored as bytes in decimal number.

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

  Specifies the media type of the entity text sent to the recipient.

Response Body

The response body is in JSON format. For details, see the definition of each API in the following sections.

1.2.3 Message Body Parameter Conventions

In the message body, variable parts in the URI are marked using angle brackets (<>).

NOTE

The URI cannot end with a slash (/).

1.3 HTTPS Methods

This section describes the common methods supported by the standard HTTPS protocol.

Table 1-3 HTTPS methods

Method	Function
POST	Creates resources, for example, customers.
DELETE	Deletes resources, for example, customers.
PUT	Updates resources as a whole, for example, updates customers as a whole. When a resource is updated using the PUT method, if some attributes of the resource in the request are not transferred, these attributes are left empty. In this case, the method is used for replacing resource attributes as a whole.
PATCH	Updates resources partially, for example, updates customers partially. When a resource is updated using the PATCH method, only the resource attributes transferred in the request are modified. Other attributes are not affected. If a resource to be updated is of the array type, all the resource attributes need to be transferred for update.
GET	Queries resources, for example, customer information.

2 Machine-Machine Interface, AK/SK-based Authentication Interface

Scenario

The API Fabric generates a token.

URL: https://*Domain Address*/apigovernance/api/oauth/tokenByAkSk

NOTE

Replace *Domain Address* with the actual address or domain name of the CEC.

For example, in the Huawei public cloud production environment, replace *Domain Address* with **service.besclouds.com**. The invoking URL is **https://service.besclouds.com/apigovernance/api/oauth/tokenByAkSk**.

Request Header

{

Content-Type: application/json

X-Token-Expire:600

}

NOTE

X-Token-Expire indicates the token expiration time, which is set to **600**.

Request Parameters

{

"app_key": "**xxxxxxxxxxxxxxxxx**",

"app_secret": "**yyyyyyyyyyyyyyyyyy**"

}

NOTE

app_key indicates the app ID, and **app_secret** indicates the key. The two values are fixed.

Response Parameters

```
{
"AccessToken": "zzzzzzzzzzzzzzzzzz",
"ApplyType": "Bearer",
"CreateTime": "1545650171",
"Expires": "600",
"Scope": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
"AppKey": "xxxxxxxxxxxxxxxxx",
"UserID": "Anonymous"
}
```

NOTE

AccessToken indicates the token of the API.

3 Third-party Request Authentication

3.1 Common Parameter Structure

3.1.1 applyTokenReqBody

Table 3-1 applyTokenReqBody Parameters

Seq.	Name	Type	Mandatory	Description
1.1	userId	string	True	User ID
1.2	userName	string	True	User Name
1.3	channelId	string	True	Channel ID
1.4	locale	string	True	Language information

3.1.2 applyTokenRspBody

Table 3-2 applyTokenRspBody Parameters

Seq.	Name	Type	Mandatory	Description
1.1	token	string	True	Specifies the generated token. The value of this parameter is the same as the value of the ccmessaging-token header field in subsequent requests.
1.2	resultCode	string	True	Result code
1.3	resultDesc	string	False	Result description

3.2 POST /ccmessaging/applyToken

Description

Third-party request, which generates a client access token.

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/applyToken.For example,the domain name is service.besclouds.com.

Request

Table 3-3 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	x-app-key	string	True	The same as app_key, which uniquely identifies a user, Configuring the ApiFabric Tool

Seq.	Name	Type	Mandatory	Description
2	Authorization	string	True	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Table 3-4 Body Parameters

Seq.	Name	Type	Mandatory	Description
1	reqBody	1.1.1	True	reqBody

Response

- **Status code: 200**

  1.1.2
- **Status code: 400**

  Incorrect request:Check the request path and parameters.
- **Status code: 401**

  Unauthorized:1.Check whether you have purchased related services.2.Contact the customer service personnel to check the current status of your account.
- **Status code: 404**

  Request not found: Check the request path.
- **Status code: 500**

  Service failure:Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description:

  URL:

  Request Parameter:

  { "userId": "XXXXXXXXX", "userName": "XXXXXXXX", "channelId": "XXXXXXXXXXXX", "locale": "zh_CN" }

  Response Parameter:

```
{ "token": "\"eyJjcmVhdGVUaW1lI\"" }
```

4 Query Queue Information

4.1 Common Parameter Structure

4.1.1 ReqBody

Table 4-1 ReqBody Parameters

Seq.	Name	Type	Mandatory	Description
1.1	userId	string	True	Unique ID of a customer
1.2	channelConfigId	string	True	Channel configuration ID
1.3	channel	string	True	Channel Type

4.1.2 RspMsg200

Table 4-2 RspMsg200 Parameters

Seq.	Name	Type	Mandatory	Description
1.1	position	integer	False	Position of the call in the queue.
1.2	totalWaitTime	integer	False	Accumulated queuing duration of a call.

Seq.	Name	Type	Mandatory	Description
1.3	currentDeviceWaitTime	integer	False	Actual waiting duration of the current skill queue.
1.4	configMaxcWaitTime	integer	False	Maximum waiting duration.
1.5	longestWaitTime	integer	False	Maximum queuing duration of a call in a queue.
1.6	estimateWaitTime	integer	False	Estimated call waiting duration of the corresponding queue.
1.7	resultCode	string	True	Return code for querying queue information. 0: success; other values: failure. For details about the failure cause, see the interface description document.

4.2 POST /ccmessaging/queryQueueInfo

Description

Query queue information for customer queuing reminders.

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/queryQueueInfo.For example,the domain name is service.besclouds.com.

Request

Table 4-3 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	False	ccmessagign-token generated by the applyToken interface
2	x-app-key	string	False	The same as app_key, which uniquely identifies a user,configuration on the API Fabric
3	Authorization	string	False	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Table 4-4 Body Parameters

Seq.	Name	Type	Mandatory	Description
1	reqBody	2.1.1	True	Request body

Response

- **Status code: 200**

  2.1.2

- **Status code: 400**

  Incorrect request:Check the request path and parameters.

- **Status code: 401**

  Unauthorized:1.Check whether you have purchased related services.2.Contact the customer service personnel to check the current status of your account.

- **Status code: 404**

  Request not found: Check the request path.

- **Status code: 500**

  Service failure:Check the values of the parameters in the request.

Error Code

None.

5 Proactively Poll Information Sent by Agents

5.1 Common Parameter Structure

5.1.1 downLlinkMsgArr

Table 5-1 downLlinkMsgArr Parameters

Seq.	Name	Type	Mandatory	Description
1.1	channel	string	True	Message Channel Type,Such as WEB.
1.2	content	string	True	Content of the MT message sent to the customer.
1.3	from	string	True	ID of the message sender.
1.4	mediaType	string	True	Message media type, for example, text type (TEXT), image (IMAGE), video (VIDEO), voice (AUDIO), and location (LOCATE).
1.5	senderNickname	string	False	Nickname of the message sender.

Seq.	Name	Type	Mandatory	Description
1.6	sourceType	string	True	Message sender, for example, agent (AGENT), robot (ROBOT), and system (SYSTEM).
1.7	timestamp	string	False	Message sending timestamp.
1.8	to	string	True	Message recipient ID.
1.9	simQuestions	string	True	Similarity Question Content.
1.10	interIdx	string	True	Number of interactions. The value is the interidx value returned by the last ODFS.

5.1.2 pollRsp

Table 5-2 pollRsp Parameters

Seq.	Name	Type	Mandatory	Description
1.1	resultDesc	string	False	Description of the returned result.
1.2	downlinkMessages	3.1.1	True	Array of messages sent to users
1.3	resultCode	string	True	Message receiving result code on the client. The value 0 indicates success, and other values indicate failure.

5.2 GET /ccmessaging/poll

Description

Receiving channel messages (pull mode)

Method

GET

URI

https://Domain Name/apiaccess/ccmessaging/poll.For example,the domain name is service.besclouds.com.

Request

Table 5-3 Query Parameters

Seq.	Name	Type	Mandatory	Description
1	receiverId	string	False	Recipient ID
2	channel	string	False	Recipient Channel Type

Table 5-4 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	True	ccmessagign-token generated by the applyToken interface
2	x-app-key	string	True	The same as app_key, which uniquely identifies a user,configuration on the API Fabric
3	Authorization	string	True	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Response

- **Status code: 200**

  3.1.2

- **Status code: 400**

Incorrect request:Check the request path and parameters.

- **Status code: 401**

  Unauthorized:1.Check whether you have purchased related services.2.Contact the customer service personnel to check the current status of your account.

- **Status code: 404**

  Request not found: Check the request path.

- **Status code: 500**

  Service failure:Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description: Message Example

  URL: Domain Address/apiaccess/ccmessaging/poll?receiverId=XXXXXXXXXXXXXXXXXXXXXXXX&channel=WEB

  Request Header:

  ccmessaging-token: XXXXXXXXX

  Response Parameters:

  { "resultCode": "0", "resultDesc": "query user messages success.", "downlinkMessages": [{ "from": "1011577355744904195", "channel": "WEB", "to": "session_fixed_user_fixed", "controlType": "CHAT", "mediaType": "TEXT", "content": "Welcome,welcome", "simQuestions": null, "senderNickname": "test", "senderAvatar": "1", "timestamp": 1619590480756, "sourceType": "ROBOT", "callId": null, "robotGender": "MALE" }, { "from": "1011577355744904195", "channel": "WEB", "to": "session_fixed_user_fixed", "controlType": "CHAT", "mediaType": "TEXT", "content": "Transfer to manual labor", "simQuestions": null, "senderNickname": "test", "senderAvatar": "1", "timestamp": 1619590482098, "sourceType": "ROBOT", "callId": null, "robotGender": "MALE" }, { "from": "1011577355744904195", "channel": "WEB", "to": "session_fixed_user_fixed", "controlType": "CHAT", "mediaType": "TEXT", "content": "Connected Agent", "simQuestions": null, "senderNickname": "System", "senderAvatar": null, "timestamp": 1619590484115, "sourceType": "SYSTEM", "callId": null, "robotGender": null }] }

6 Submit Customer Satisfaction Evaluation Results

6.1 Common Parameter Structure

6.1.1 RspMsg

Table 6-1 RspMsg Parameters

Seq.	Name	Type	Mandatory	Description
1.1	resultCode	string	True	Return Code
1.2	resultDesc	string	False	Response Description

6.1.2 satisficationInfo

Table 6-2 satisficationInfo Parameters

Seq.	Name	Type	Mandatory	Description
1.1	channel	string	True	Channel Type
1.2	userId	string	True	User ID
1.3	evaluation	string	True	Satisfaction level (1-5)

Seq.	Name	Type	Mandatory	Description
1.4	comment	string	True	Satisfaction comment, which contains less than 512 characters
1.5	channelConfigId	string	True	Channel ID

6.2 POST /ccmessaging/saveSatisfaction

Description

Saves the satisfaction evaluation submitted by users

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/saveSatisfaction.For example,the domain name is service.besclouds.com.

Request

Table 6-3 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	True	ccmessagign-token generated by the applyToken interface
2	x-app-key	string	True	The same as app_key, which uniquely identifies a user, Configuring the ApiFabric Tool
3	Authorization	string	True	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Table 6-4 Body Parameters

Seq.	Name	Type	Mandatory	Description
1	satisfication	4.1.2	True	Satisfaction evaluation information

Response

- **Status code: 200**

  4.1.1

- **Status code: 400**

  Incorrect request:Check the request path and parameters.

- **Status code: 401**

  Unauthorized:1.Check whether you have purchased related services.2.Contact the customer service personnel to check the current status of your account.

- **Status code: 404**

  Request not found: Check the request path.

- **Status code: 500**

  Service failure:Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description:

  URL:

  Request Parameter:

  { "channel": "WEB", "userId": "XXXXXXXX", "evaluation": "5", "comment": "Satisfied", "channelConfigId": "XXXXXXXXX" }

  Response Parameter:

  { "returnCode": "0" }

- Scene description:demo

  URL:

  Request Parameter:

  { "channel": "WEB", "userId": "XXXXXXXX", "evaluation": "5", "comment": "Satisfied", "channelConfigId": "XXXXXXXXX" }

  Response Parameter:

  { "returnCode": "0" }

7 Send Message

7.1 Common Parameter Structure

7.1.1 senMsgRsp

Table 7-1 senMsgRsp Parameters

Seq.	Name	Type	Mandatory	Description
1.1	resultDesc	string	False	Result Code Description
1.2	resultCode	string	True	Return code of the interface. The value 0 indicates success, and other values indicate failure.

7.1.2 sendMsgReq

Table 7-2 sendMsgReq Parameters

Seq.	Name	Type	Mandatory	Description
1.1	channel	string	True	Type of the channel through which the client sends messages.For example, access through the web channel (WEB).
1.2	content	string	True	Message content, which is determined by mediaType. For example, if mediarrType is set to TXT, content indicates the text content. For media such as audio and video, the value of content is the same as that of fileId. The file download interface uses this fileId to download files.
1.3	controlType	string	True	Message control type, such as connect(CONNECT) chat (CHAT) and disconnect (DISCONNECT),Transfer to manual service(TRANS2AGENT),Entering(INPUT)
1.4	from	string	True	ID of the user who sends a message through the channel.
1.5	mediaType	string	True	Message media type,such as text type (TEXT)
1.6	senderNickname	string	False	Nickname of the message sender.
1.7	sourceType	string	True	Message source type.

Seq.	Name	Type	Mandatory	Description
1.8	timestamp	string	False	Time when a message is sent.
1.9	to	string	True	Channel configuration ID.
1.10	userAgent	string	False	User client type, which is obtained from the UserAgent.
1.11	senderAvatar	string	False	Avatar of the message sender. The WeChat avatar URL is transmitted through the WeChat channel.

7.2 POST /ccmessaging/send

Description

Sending channel messages, including connection messages, sending input status messages, session messages, transfer to manual services, and disconnection messages

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/send.For example,the domain name is service.besclouds.com.

Request

Table 7-3 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	True	ccmessagign-token generated by the applyToken interface

Seq.	Name	Type	Mandatory	Description
2	x-app-key	string	True	The same as app_key, which uniquely identifies a user, Configuring the ApiFabric Tool
3	Authorization	string	True	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Table 7-4 Body Parameters

Seq.	Name	Type	Mandatory	Description
1	reqBody	5.1.2	True	Request body

Response

- **Status code: 200**

  5.1.1
- **Status code: 400**

  Incorrect request:Check the request path and parameters.
- **Status code: 401**

  Unauthorized:1.Check whether you have purchased related services.2.Contact the customer service personnel to check the current status of your account.
- **Status code: 404**

  Request not found: Check the request path.
- **Status code: 500**

  Service failure:Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description: Connecting Robots

  URL:

Request Parameter:

{ "content": "", "controlType": "CONNECT", "from": "session_1619573111751_user_1619573111751", "mediaType": "TEXT", "senderNickname": "user_1619573111751", "sourceType": "CUSTOMER", "timestamp": "1619573122352", "to": "1011577355744904195", "userAgent": null, "senderAvatar": null, "channel": "WEB" }

Response Parameter:

{ "resultCode": "0", "resultDesc": "connect to robot success." }

- Scene description: Talk to Bot Request

  URL:

  Request Parameter:

  { "content": "Version", "controlType": "CHAT", "from": "session_1619573111751_user_1619573111751", "mediaType": "TEXT", "senderNickname": "user_1619573111751", "sourceType": "CUSTOMER", "timestamp": "1619573123199", "to": "1011577355744904195", "userAgent": null, "senderAvatar": null, "channel": "WEB" }

  Response Parameter:

  { "resultCode": "0", "resultDesc": "Send message to robot successfully" }

- Scene description: Robot-to-manual request

  URL:

  Request Parameter:

  { "content": "", "controlType": "TRANS2AGENT", "from": "session_1619573111751_user_1619573111751", "mediaType": "TEXT", "senderNickname": "user_1619573111751", "sourceType": "CUSTOMER", "timestamp": "1619573129953", "to": "1011577355744904195", "userAgent": null, "senderAvatar": null, "channel": "WEB" }

  Response Parameter:

  { "resultCode": "0", "resultDesc": "transfer from robot to agent success." }

- Scene description: Disconnection request

  URL:

  Request Parameter:

  { "content": "", "controlType": "DISCONNECT", "from": "session_1619573842490_user_1619573842490", "mediaType": "TEXT", "senderNickname": "user_1619573842490", "sourceType": "CUSTOMER", "timestamp": "1619573854156", "to": "1011577355744904195", "userAgent": null, "senderAvatar": null, "channel": "WEB" }

  Response Parameter:

  { "resultCode": "0", "resultDesc": "Disconnect robot successfully" }

8 Send Files by Customer

8.1 Common Parameter Structure

8.1.1 ReqBody

Table 8-1 ReqBody Parameters

Seq.	Name	Type	Mandatory	Description
1.1	fileType	string	True	File type. The value can be jpg,png,bpm,mp3,aac, mp4,txt,csv.
1.2	fileStream	string	True	Base64-encoded file stream
1.3	channel	string	True	Channel type,Such as WEB

8.1.2 RspBody

Table 8-2 RspBody Parameters

Seq.	Name	Type	Mandatory	Description
1.1	objectKey	string	False	objectKey (file server storage key)

Seq.	Name	Type	Mandatory	Description
1.2	resultCode	string	True	Result code of the file upload. The value 0 indicates that the file upload is successful. Other values indicate that the file upload fails.
1.3	resultDesc	string	False	Response Description

8.2 POST /ccmessaging/uploadFileStream

Description

Send a file

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/uploadFileStream.For example,the domain name is service.besclouds.com.

Request

Table 8-3 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	True	ccmessagign-token generated by the applyToken interface
2	x-app-key	string	True	The same as app_key, which uniquely identifies a user

Seq.	Name	Type	Mandatory	Description
3	Authorization	string	True	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Table 8-4 Body Parameters

Seq.	Name	Type	Mandatory	Description
1	reqBody	6.1.1	True	Request body

Response

- **Status code: 200**

  6.1.2
- **Status code: 400**

  Incorrect request:Check the request path and parameters.
- **Status code: 401**

  Unauthorized:1.Check whether you have purchased related services.2.Contact the customer service personnel to check the current status of your account.
- **Status code: 404**

  Request not found: Check the request path.
- **Status code: 500**

  Service failure:Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description:

  URL:

  Request Parameter:

  { "fileType": "jpg", "channel": "WEB", "fileStream": "/9j/4AAQSkZJRgABAQEAYABgAAD/2wBDAAMCAgMCAgMDAwMEAwMEBQgFBQQEBQoHBwYIp3+7j8a/PP/gqXP5/7QXh9tu3/AIpi34z/ANPV3SpXfxCP/9k=" }

Response Parameter:

{ "returnCode": "0", "objectKey": "44992392-2078-4caa-af3f-16cc328db296" }

9 Receive Document by Customer

9.1 Common Parameter Structure

9.1.1 ReqBody

Table 9-1 ReqBody Parameters

Seq.	Name	Type	Mandatory	Description
1.1	fileId	string	True	File ID for file download, which is carried in the poll interface
1.2	channel	string	True	Download Channel Type
1.3	fileType	string	True	Downloaded File Type

9.1.2 RspBody

Table 9-2 RspBody Parameters

Seq.	Name	Type	Mandatory	Description
1.1	fileStream	string	False	Base64-encoded file stream

Seq.	Name	Type	Mandatory	Description
1.2	resultCode	string	True	Response result code. The value 0 indicates success, and other values indicate failure.
1.3	resultDesc	string	False	Response Description

9.2 POST /ccmessaging/downloadFileStream

Description

User Accepting Files

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/downloadFileStream.For example,the domain name is service.besclouds.com.

Request

Table 9-3 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	True	ccmessagign-token generated by the applyToken interface
2	x-app-key	string	True	The same as app_key, which uniquely identifies a user,configuration on the API Fabric
3	Authorization	string	True	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Table 9-4 Body Parameters

Seq.	Name	Type	Mandatory	Description
1	reqBody	7.1.1	True	Request body

Response

- **Status code: 200**

  7.1.2
- **Status code: 400**

  Incorrect request:Check the request path and parameters.
- **Status code: 401**

  Unauthorized:1.Check whether you have purchased related services.2.Contact the customer service personnel to check the current status of your account.
- **Status code: 404**

  Request not found: Check the request path.
- **Status code: 500**

  Service failure:Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description:demo

  URL:

  Request Parameter:

  { "fileType": "jpg", "channel": "WEB", "fileId": "XXXXXXXXXXXXXXXX" }

  Response Parameter:

  { "fileStream": "/9j/4AAQSkZJRgABAQEAYABgAAD/2wBDAAMCAgMCAgMDAwMEAwMEBQgFBQQEBQoHBwYIDAoMDAsKCwsNDhIQDQ4RDgsLEBYQERMUFRUVDA8XGBYUGBIUFRT/2wBDAQMEBAUEBQkFBQkUDQsNFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBT/wAARCAA9ADwDASIAAhEBAxEB/8QAHwAAAQUBAQEBAQEAAAAAAAAAAAECAwQFBgcI" }

10 Associate Common Phrases

10.1 Common Parameter Structure

10.1.1 ReqBody

Table 10-1 ReqBody Parameters

Seq.	Name	Type	Mandatory	Description
1.1	strContent	string	True	Message content
1.2	locale	string	True	Language
1.3	channel	string	True	Channel type, for example, WEB.

10.1.2 RspBody

Table 10-2 RspBody Parameters

Seq.	Name	Type	Mandatory	Description
1.1	phrases	string	True	Array of similar questions.

Seq.	Name	Type	Mandatory	Description
1.2	resultCode	string	True	Return code of the response. The value 0 indicates success, and other values indicate failure.
1.3	resultDesc	string	False	Response Description

10.2 POST /ccmessaging/queryPhraseByKeyword

Description

Associate common phrases based on keywords entered by users.

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/queryPhraseByKeyword.For example,the domain name is service.besclouds.com.

Request

Table 10-3 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	True	ccmessagign-token generated by the applyToken interface
2	x-app-key	string	False	The same as app_key, which uniquely identifies a user,configuration on the API Fabric

Seq.	Name	Type	Mandatory	Description
3	Authorization	string	False	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Table 10-4 Body Parameters

Seq.	Name	Type	Mandatory	Description
1	reqBody	8.1.1	True	Request body

Response

- **Status code: 200**

  8.1.2
- **Status code: 400**

  Incorrect request:Check the request path and parameters.
- **Status code: 401**

  Unauthorized:1.Check whether you have purchased related services.2.Contact the customer service personnel to check the current status of your account.
- **Status code: 404**

  Request not found: Check the request path.
- **Status code: 500**

  Service failure:Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description: Message Example

  URL:

  Request Header:

  ccmessaging-token: XXXXXXXXX

  Request Parameter:

  { "strContent": "Mesh", "locale": "zh", "channel": "WEB" }

  Response Parameter:

```
{ "question": [ "Where can I download the Grid app?", "How do I log in to the Grid app?", "What Mobile Phone Versions Does the Grid App Support?", "After logging in to the grid version of the app, you do not have the permission to scan the street to place orders and grid account menus.", "How to log in to the middle-screen grid system on a PC?" ], "resultCode": "0" }
```

11 Create Click-to-dial Call

11.1 POST /ccmessaging/v1/createClickToCall

Description

create click to call

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/v1/createClickToCall,the domain name is service.besclouds.com.

Request

Table 11-1 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	False	ccmessagign-token generated by the applyToken interface
2	x-app-key	string	False	The same as app_key, which uniquely identifies a user

Seq.	Name	Type	Mandatory	Description
3	Authorization	string	False	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Table 11-2 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	channel	string	True	channel type ('WEB')

1.2	mediaAbility	string	True	mediaAbility:
				0: audio call;
				1: video call;

Response

- **Status code: 200**

Table 11-3 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	resultCode	string	False	result Code
1.2	resultDesc	string	False	result description

- **Status code: 400**

  Incorrect request: Check the request path and parameters.
- **Status code: 401**

  Unauthorized: 1. Check whether you have purchased related services. 2. Contact customer service personnel to check the current status of your account.
- **Status code: 404**

  Request content not found: Check the path of the request.
- **Status code: 500**

  Business failure: Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description:create voice call

  URL:create voice call

  Request Header:

  ccmessaging-token:eyJjcmVhdGVUaW1lIjoxNjA3MDc0NzIxMTc4 Authorization:Bearer dbb011d5df2b9466d20b0b1845bbb33a x-app-key:6e9d9cd8-e135-4ab8-9f78-88e6b152c2f6 Content-Type:application/json

  Request Parameter:

  { "channel":"WEB", "mediaAbility":"0" }

  Response Parameter:

  { "resultCode":"0", "resultDesc:":null }

12 Get Click-to-dial Event

12.1 GET /ccmessaging/v1/getClickToCallEvents

Description

get ClickToCall Events

Method

GET

URI

https://Domain Name/apiaccess/ccmessaging/v1/getClickToCallEvents,the domain name is service.besclouds.com.

Request

Table 12-1 Query Parameters

Seq.	Name	Type	Mandatory	Description
1	channel	string	True	channel type" WEB"

Table 12-2 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	True	ccmessagign-token generated by the applyToken interface
2	x-app-key	string	True	The same as app_key, which uniquely identifies a user
3	Authorization	string	True	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Response

- **Status code: 200**

Table 12-3 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	resultCode	string	False	result Code
1.2	resultDesc	string	False	result description
1.3	eventId	string	False	event Id
1.4	content	object	False	content
1.4.1	causeDesc	string	False	cause description
1.4.2	causeId	string	False	cause id
1.4.3	accessCode	string	False	access Code
1.4.4	domain	string	False	webRTC domain
1.4.5	gwAddresses	array	False	webRTC gateway address
1.4.6	clickToCallCaller	string	False	Caller

- **Status code: 400**

  Incorrect request: Check the request path and parameters.

- **Status code: 401**

  Unauthorized: 1. Check whether you have purchased related services. 2. Contact customer service personnel to check the current status of your account.
- **Status code: 404**

  Request content not found: Check the path of the request.
- **Status code: 500**

  Business failure: Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description:get create call event

  URL:get create call event

  Request Header:

  ccmessaging-token: x-app-key: Authorization:

  Request Parameter:

  /apiaccess/ccmessaging/getClickToCallEvents?channel=WEB

  Response Parameter:

  { resultCode:"0", resultDesc:"Call connected", "eventId": 168101, "content": { "domain":"xxx" "gwAddresses":["xx1","xx2"] "accessCode": "179080000537636" "clickToCallCaller":"AnonymousCard" } }

13 Check Whether Support Click-to-dial

13.1 GET /ccmessaging/v1/checkClickToCallSupport

Description

check if it supports 'click to call'

Method

GET

URI

https://Domain Name/ccmessaging/v1/checkClickToCallSupport,the domain name is service.besclouds.com.

Request

Table 13-1 Query Parameters

Seq.	Name	Type	Mandatory	Description
1	channel	string	True	channel type WEB

Table 13-2 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	True	ccmessagign-token generated by the applyToken interface
2	x-app-key	string	True	The same as app_key, which uniquely identifies a user
3	Authorization	string	True	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Response

- **Status code: 200**

Table 13-3 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	webRTCSupported	boolean	False	if it supports WebRTC
1.2	clickToCallSupported	boolean	False	if it supports 'click to call'
1.3	resultCode	string	False	resultCode
1.4	resultDesc	string	False	resultDesc

- **Status code: 400**

  Incorrect request: Check the request path and parameters.

- **Status code: 401**

  Unauthorized: 1. Check whether you have purchased related services. 2. Contact customer service personnel to check the current status of your account.

- **Status code: 404**

  Request content not found: Check the path of the request.

- **Status code: 500**

  Business failure: Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description:check if it supports click to call

  URL:check if it supports click to call

  Request Header:

  ccmessaging-token x-app-key Authorization

  Request Parameter:

  /apiaccess/ccmessaging/checkClickToCallSupport?channel=WEB

  Response Parameter:

  { "resultCode":"0", "resultDesc": "", "webRTCSupported":true, "clickToCallSupported":true}

14 Release Click-to-dial Call

14.1 POST /ccmessaging/v1/dropClickToCall

Description

drop call

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/v1/dropClickToCall,the domain name is service.besclouds.com.

Request

Table 14-1 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	True	ccmessagign-token generated by the applyToken interface
2	x-app-key	string	True	The same as app_key, which uniquely identifies a user

Seq.	Name	Type	Mandatory	Description
3	Authorization	string	True	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Table 14-2 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	channel	string	True	channel type ('WEB')

Response

- **Status code: 200**

Table 14-3 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	resultCode	string	False	result Code
1.2	resultDesc	string	False	result description

- **Status code: 400**

  Incorrect request: Check the request path and parameters.

- **Status code: 401**

  Unauthorized: 1. Check whether you have purchased related services. 2. Contact customer service personnel to check the current status of your account.

- **Status code: 404**

  Request content not found: Check the path of the request.

- **Status code: 500**

  Business failure: Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description:drop click to call

  URL:drop click to call

  Request Header:

  'Content-Type': 'application/json', 'x-app-key': appKey, 'Authorization': fabric.token, 'ccmessaging-token': ccmessaging.token

  Request Parameter:

  { "channel": "WEB" }

  Response Parameter:

  { "resultCode":"0", "resultDesc": "" }

15 Submit Offline Messages by Customer

15.1 POST /ccmessaging/doLeaveMessage

Description

Interface for saving user messages

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/doLeaveMessage.For example,the domain name is service.besclouds.com.

Request

Table 15-1 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	ccmessaging-token	string	True	ccmessagign-token generated by the applyToken interface
2	x-app-key	string	True	The same as app_key, which uniquely identifies a user

Seq.	Name	Type	Mandatory	Description
3	Authorization	string	True	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Table 15-2 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	callId	string	True	User ID
1.2	caller	string	True	Number of a user who leaves a message.
1.3	leaveMessage	string	True	User message
1.4	channel	string	True	Channel ID (Web)

Response

- **Status code: 200**

Table 15-3 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	resultCode	string	True	Return Code
1.2	resultDesc	string	False	Response Description

- **Status code: 400**

  Incorrect request:Check the request path and parameters.
- **Status code: 401**

  Unauthorized:1.Check whether you have purchased related services.2.Contact the customer service personnel to check the current status of your account.
- **Status code: 404**

  Request not found: Check the request path.
- **Status code: 500**

Service failure:Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description:

  URL:

  Request Parameter:

  { "callId":"111111222111", "caller":"13866667777", "leaveMessage":"Please call me", "channel":"WEB" }

  Response Parameter:

  { "resultCode": "0", "resultDesc": "Save leaveMessage success!" }

16 Create an Anonymous Call

16.1 POST /ccmessaging/v1/mobileagent/createcall

Description

An operator logs in to the app and invokes the createcall interface provided by the Chat to create an anonymous outbound call task and initiates an anonymous call through the integrated SDK.

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/v1/mobileagent/createcall.For example,the domain name is service.besclouds.com.

Request

Table 16-1 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	X-APP-Key	string	True	The same as app_key, which uniquely identifies a user, Configuring the ApiFabric Tool

Seq.	Name	Type	Mandatory	Description
2	Authorization	string	True	Authentication field. The value format is as follows: Bearer +AccessToken in the return values of the tokenByAKSK interface (there is a space after Bearer)

Table 16-2 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	called	string(32)	True	Called number
1.2	callData	string(1,024)	False	Call data
1.3	callerPresent	string(32)	True	Calling number, which is displayed by subscribers and customer managers. If the calling number is not contained, the default calling number of the platform is used.

Response

- **Status code: 200**

Table 16-3 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	resultCode	string	True	Return Code. 0:success 30010000001:This feature is not enabled and cannot be used 30010000002:No SIP server information is found 30010000003:Input parameter verification failed 30010010001:Failed to add the outbound call task 30010010002:The VDN is unavailable 30010010003:The VDN configuration is incorrect, and the record path fails to be generated 30010019999:CCM other exception 30011010001:Failed to create a call without waiting for the call setup success event 30011019999:CCUCS other exception 30010009999:Chat other exception
1.2	resultDesc	string	False	Failure Cause Description
1.3	resultData	object	False	Response Data Structure
1.3.1	callSerialNo	string(50)	False	Call sequence number, which uniquely identifies an outgoing call
1.3.2	sipServiceAddress	string	True	IP address of the SIP service. The value is ip:port

Seq.	Name	Type	Mandatory	Description
1.3.3	accesscode	string	True	Used for anonymous VoIP calls. The value is the access code and UVID

- **Status code: 400**

  Incorrect request:Check the request path and parameters.

- **Status code: 401**

  Unauthorized:1.Check whether you have purchased related services.2.Contact the customer service personnel to check the current status of your account.

- **Status code: 404**

  Request not found: Check the request path.

- **Status code: 500**

  Service failure:Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description:Invoke the chat interface to create an outbound call task

  URL:

  Request Header:

  { "X-APP-Key": "b5cb838d-8b7a-4dc8-8aab-0b91edc7ec7b", "Authorization": "Bearer 1fea3651a3a09fa8489054bc3e1d6a15", "Content-Type": "application/json" }

  Request Parameter:

  { "called":"88884432", "callData":"1", "callerPresent":"95599" }

  Response Header:

  connection →keep-alive content-encoding →gzip content-length →130 content-type →application/json;charset=UTF-8 date →Wed, 12 May 2021 08:52:40 GMT server →openresty

  Response Parameter:

  { "resultCode": "0", "resultDesc": "Success", "resultData": { "sipServiceAddress": "10.137.203.215:5063", "accesscode": "179080024609291" } }

17 Submit Customer Satisfaction with Robots

17.1 POST /ccmessaging/v1/robot/feedbacksatisfaction

Description

Robot Response Satisfaction Evaluation

Method

POST

URI

https://Domain Name/apiaccess/ccmessaging/v1/robot/feedbacksatisfaction.For example,the domain name is service.besclouds.com.

Request

Table 17-1 Header Parameters

Seq.	Name	Type	Mandatory	Description
1	x-app-key	string	True	appKey field, which indicates the user ID.

Seq.	Name	Type	Mandatory	Description
2	Authorization	string	True	Authentication field. The content format is Bearer+tokenByAKSK. The value of AccessToken in the return value of the API contains a space.
3	ccmessaging-token	string	True	ccmessign-token generated by the applyToken API

Table 17-2 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	feedback	integer	True	Satisfaction with feedback. 1: satisfied; 0: dissatisfied
1.2	feedbackContent	string(64)	False	When the value is 0, the value cannot be empty. The maximum length is 64 bytes.
1.3	from	string	True	ID of the message sender.
1.4	interIdx	string	True	Number of interactions. The value is the interidx value returned by the last ODFS.

Response

- **Status code: 200**

Table 17-3 Body Parameters

Seq.	Name	Type	Mandatory	Description
1.1	resultCode	string	True	0:success 30000000008:feedbackContent can not be empty,when feedback is 0 10010000002:Access failed because of flow control 30010009999:request chat error 10010000001:Authentication failed
1.2	resultDesc	string	False	Failure Cause Description

- **Status code: 400**

  Incorrect request: Check the request path and parameters.
- **Status code: 401**

  Unauthorized: 1. Check whether you have purchased related services. 2. Contact customer service personnel to check your account status.
- **Status code: 404**

  Requested content not found: Please check the requested path.
- **Status code: 500**

  Service failure: Check the values of the parameters in the request.

Error Code

None.

Message Example

- Scene description:Robot Response Satisfaction Evaluation

  URL:Request success packet

  Request Header:

  { "ccmessaging-token":"eyJjcmVhdGVUaW1lIjoxNjIyNjI2Mjc3NDA4LCJ0ZW5hbnRJZCI6IjIwMjEwMjE5MzE5NiIsInJhbmRvbU51bWJlciI6IjZDRjEyOTE0OTQ3QzY1NzUwRUI4RDRBNENDODZEQTJGRTE4RjhBOTVERjcwMzA1QyIsIm1hcFNlcnZpY2UiOiJudWxsIiwidXNlck5hbWUiOiIyMzEyMzIxIiwibG9jYWxlIjoiemgiLCJ1c2VySWQiOiIxMTExMTExIiwiY2hhbm5lbElkIjoiMjAyMTAyMTkzMTk2ODY5NzA4In0%3D", "x-app-key":"ea5c29a3-ea45-4b79-a449-2a3dbeb4c32c", "Authorization":"Bearer 57042f85aeabd63562f2e8339968c4e0", "Content-Type":"application/json" }

Request Parameter:

{ "feedback":0, "feedbackContent":"dissatisfied", "from":"1111111", "interIdx":"2" }

Response Parameter:

{ "resultCode": "0", "resultDesc": "Send message to robot successfully" }