**GEA project interface definition specification**

**Project name： GEA security**

**Vservion： VX.X**

**Date：**

Document revision record

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Standard revision history record | | | | | | | | | |
| Version | Data | | Reviser | Revision page | | | Revise the content | | |
| 1.0 | 2017-01-03 | | Wu Xiaobo |  | | | New document | | |
| 1.1 | 2017-01-19 | | Wu Xiaobo |  | | | Modify the DeviceInfo object to increase the activation mail interface | | |
| 1.2 | 2017-01-22 | | Wu Xiaobo |  | | | Add picture captcha related interface | | |
| 1.3 | 2018-06-22 | | Yu Shaochen |  | | | Add user acceptance privacy policy interface | | |
| 1.4 | 2018-07-10 | | Jia Shanshan |  | | | Modify the wrong url in the device class partial interface request example | | |
|  |  | |  |  | | |  | | |
|  |  | |  |  | | |  | | |
| 会 签 | | | | | | | | | |
| Business entity | | Signing the opinion | | | | Signature | | | Date |
|  | |  | | | |  | | |  |
|  | |  | | | |  | | |  |
|  | |  | | | |  | | |  |
|  | |  | | | |  | | |  |
|  | |  | | | |  | | |  |
|  | |  | | | |  | | |  |
|  | |  | | | |  | | |  |
| Establishment | |  | | | Check | | |  | |
| Audit | |  | | | Approve | | |  | |

**Directory**

[1 Introduction 9](#_Toc521674716)

[1.1 Executive summary 9](#_Toc521674717)

[1.2 Writing purpose 9](#_Toc521674718)

[1.3 Terms and definitions 9](#_Toc521674719)

[1.4 Reference material 9](#_Toc521674720)

[1.5 Special Symbol Format Description 9](#_Toc521674721)

[2 Public description 10](#_Toc521674722)

[2.1 Basic introduction 10](#_Toc521674723)

[2.2 Calling process 10](#_Toc521674724)

[2.3 Access address 11](#_Toc521674725)

[2.4 Interface common part 11](#_Toc521674726)

[2.5 Signature authentication 13](#_Toc521674727)

[2.5.1 Instructions 13](#_Toc521674728)

[2.5.2 Introduction to parameters 13](#_Toc521674729)

[2.5.3 Algorithm 13](#_Toc521674730)

[2.6 Security of user password 13](#_Toc521674731)

[2.6.1 Safety instructions 13](#_Toc521674732)

[2.6.2 Password flow 14](#_Toc521674733)

[2.6.3 Cipher encryption and decryption algorithm 14](#_Toc521674734)

[2.7 Internationalization 15](#_Toc521674735)

[2.8 Data type qualification 15](#_Toc521674736)

[2.8.1 Field type description 15](#_Toc521674737)

[2.8.2 Null value description 15](#_Toc521674738)

[3 User interface 17](#_Toc521674739)

[3.1User registration 17](#_Toc521674740)

[3.1.1 Interface definition 17](#_Toc521674741)

[3.1.2 Return code 18](#_Toc521674742)

[3.1.3 Request example 19](#_Toc521674743)

[3.2 User login 20](#_Toc521674744)

[3.2.1 Interface definition 20](#_Toc521674745)

[3.2.2 Return code 21](#_Toc521674746)

[3.2.3 Request example 21](#_Toc521674747)

[3.3 Send activation 22](#_Toc521674748)

[3.3.1 Interface definition 22](#_Toc521674749)

[3.3.2 Return code 23](#_Toc521674750)

[3.3.3 Request example 23](#_Toc521674751)

[3.4 Log out 24](#_Toc521674752)

[3.4.1 Interface definition 24](#_Toc521674753)

[3.4.2 Return code 24](#_Toc521674754)

[3.4.3 Request example 24](#_Toc521674755)

[3.5 Query user information 25](#_Toc521674756)

[3.5.1 Interface definition 25](#_Toc521674757)

[3.5.2 Retrun code 25](#_Toc521674758)

[3.5.3 Request example 26](#_Toc521674759)

[3.6 User information modification 27](#_Toc521674760)

[3.6.1 Interface definition 27](#_Toc521674761)

[3.6.2 Return code 27](#_Toc521674762)

[3.6.3 Request example 27](#_Toc521674763)

[3.7 Apply for reset password 28](#_Toc521674764)

[3.7.1 Interface definition 28](#_Toc521674765)

[3.7.2 Return code 29](#_Toc521674766)

[3.7.3Request example 29](#_Toc521674767)

[3.8 Get the picture verification code 30](#_Toc521674768)

[3.8.1 Interface definition 30](#_Toc521674769)

[3.8.2 Return code 30](#_Toc521674770)

[3.8.3 Request example 30](#_Toc521674771)

[4 Device class interface 31](#_Toc521674772)

[4.1 Public structure 31](#_Toc521674773)

[4.1.1 DeviceInfo 31](#_Toc521674774)

[4.1.2 AuthInfo 32](#_Toc521674775)

[4.1.3 Permission 32](#_Toc521674776)

[4.1.4 Location 32](#_Toc521674777)

[4.1.5 BaseProperty 32](#_Toc521674778)

[4.1.6 DeviceVersion 33](#_Toc521674779)

[4.1.7 Module 33](#_Toc521674780)

[4.2 Query the user device list 33](#_Toc521674781)

[4.2.1 Interface definition 33](#_Toc521674782)

[4.2.2 Return code 34](#_Toc521674783)

[4.2.3 Request example 34](#_Toc521674784)

[4.3 Binding equipment 38](#_Toc521674785)

[4.3.1 Interface definition 38](#_Toc521674786)

[4.3.2 Return code 39](#_Toc521674787)

[4.3.3 Request example 39](#_Toc521674788)

[4.4 Unbundling equipment 40](#_Toc521674789)

[4.4.1 Interface definition 40](#_Toc521674790)

[4.4.2 Return code 40](#_Toc521674791)

[4.4.3 Request example 41](#_Toc521674792)

[4.5 Modify device alias 41](#_Toc521674793)

[4.5.1 Interface definition 41](#_Toc521674794)

[4.5.2 Return code 42](#_Toc521674795)

[4.5.3 Request example 42](#_Toc521674796)

[4.6 Modify device location information 43](#_Toc521674797)

[4.6.1 Interface definition 43](#_Toc521674798)

[4.6.2 Return code 43](#_Toc521674799)

[4.6.3 Request example 43](#_Toc521674800)

[4.7 Get device alias 44](#_Toc521674801)

[4.7.1 Interface definition 44](#_Toc521674802)

[4.7.2 Return code 44](#_Toc521674803)

[4.7.3 Request example 44](#_Toc521674804)

[4.8 Obtain device location information 45](#_Toc521674805)

[4.8.1 Interface definition 45](#_Toc521674806)

[4.8.2 Return code 45](#_Toc521674807)

[4.8.3 Request example 46](#_Toc521674808)

[4.9 Obtain device details 46](#_Toc521674809)

[4.9.1 Interface definition 46](#_Toc521674810)

[4.9.2 Return code 47](#_Toc521674811)

[4.9.3 Request example 47](#_Toc521674812)

[4.10 Inquire if the device is online 49](#_Toc521674813)

[4.10.1 Interface definition 49](#_Toc521674814)

[4.10.2 Return code 49](#_Toc521674815)

[4.10.3 Request example 49](#_Toc521674816)

[5 Capability class interface 50](#_Toc521674817)

[5.1 App version query 50](#_Toc521674818)

[5.1.1 Interface definition 50](#_Toc521674819)

[5.1.2 Return code 51](#_Toc521674820)

[5.1.3 Request example 51](#_Toc521674821)

[5.2 Upload resource file 52](#_Toc521674822)

[5.2.1 Interface definition 52](#_Toc521674823)

[5.2.2 Return code 52](#_Toc521674824)

[5.2.3 Request example 53](#_Toc521674825)

[6 Appendix 54](#_Toc521674826)

[6.1 Common Error Codes 54](#_Toc521674827)

[6.2 Internationalized language code list 55](#_Toc521674828)

[6.3 Example signature algorithm 57](#_Toc521674829)

[6.4 AES encryption and decryption algorithm code example 58](#_Toc521674830)

1. Introduction
   1. Executive summary

As the basis and input of overseas GEA project UWS development for Haier U+ cloud platform.

As the basis and input for third-party application development.

* 1. Writing purpose

Provide the foundation and guidance for third-party application development.

* 1. Terms and definitions

The Haier U+ cloud platform(IoT cloud platform) provides access services and data analysis services for devices.USDK, APP development package, in line with E++ protocol, with the functions of discovering devices, configuring devices to access the Internet, and communicating with devices.

* 1. Reference material

The relevant documents and standards referred to in this design are as follows:

* Signature specification for cloud platform UWS
* Cloud platform Rest interface design specification
* Cloud platform user class interface specification
* Cloud platform composite class interface design
* Cloud platform capability class interface design
* Overseas good air interface manual
  1. Special Symbol Format Description

This document does not use other special document marks or formats.

The interface examples in this article use sample data that is not real data.

1. Public description
   1. Basic introduction

uws provides interfaces for applications to access the U+ cloud platform, and these interfaces and apis are described in the documentation. Readers of this document need to have some technical knowledge of HTTP, HTTPS, REST, RPC, and other technologies.

The uws interface uses the REST-RPC style, and the supported data format is JSON.

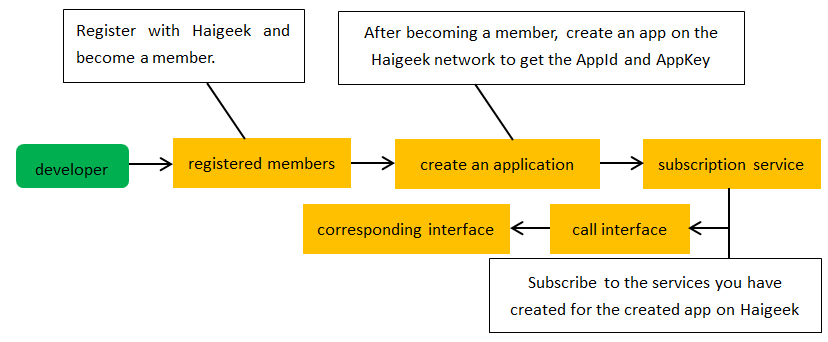
The uws interface uses HTTPS one-way certificate authentication for communication to enhance security.

uws will strongly check the parameters of the sent request. If there is any inconsistency with the interface specification, the interface call will be wrong.

uws may increase the number of interface request and response parameters when expanding interface functions in the future. The interface consumer should be able to do this.

* 1. Calling process

To use the interface in the document, the App or App Server follows the following process.



* 1. Access address

All interfaces provided in this document only support https protocol requests.

In development and joint debugging, the application developer should connect to the developer environment and connect to different development environments by configuring the application developer to access the router of the external network (setting the router's dns).

European environment domain name：**[https://uws-gea-euro.haieriot.net](https://uws-gea-euro.haieriot.net/" \t "_self)**

North American environment domain name：**[https://uws-gea-us.haieriot.net](https://uws-gea-us.haieriot.net/" \t "_self)**

* 1. Interface common part

In the application interaction with uws, the application needs to pass some fixed parameters in each request header; each response of uws also contains a fixed response code, as follows:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Input parameters that need to be passed in the request | | | | | |
| Parameter names | | Type | Location | Required | Instructions |
| appId | | String | Header | Yes | Application ID, within 40 characters,Haier U+ cloud platform globally unique. |
| appVersion | | String | Header | Yes | Apply version up to 32 characters, apply version identification. |
| clientId | | String | Header | Yes | The client ID, which is primarily used to uniquely identify the client (for example, a mobile phone). You can call usdk to get the value of the client ID. |
| sequenceId | | String | Header | Yes | Message flow (Client unique) client transaction serial number. 6-32 bit. It is defined by the client and generated by itself. It is recommended to use date + sequential numbering. |
| accessToken | | String | Header | Yes | Security token, 30 bit character. Users log on Haier U+ cloud platform, which is created by the system. Users exit Haier U+ cloud platform and the system will destroy it. When you are not logged in, access the platform interface that does not require login, and you still need to pass in this parameter. The parameter value can be null or any value (no more than 30 characters). |
| sign | | String | Header | Yes | The signature generated by signing the request is shown in chapter 2.5. |
| timestamp | | long | Header | Yes | Unix timestamp.Accurate to milliseconds. |
| language | | String | Header | Yes | An international logo that represents the language used by the client. The specific identification code is shown in the appendix. |
| timezone | | int | Header | Yes | On behalf of the time zone used by the client, uws does not support internationalization at the moment, fill in 8. |
| Content-Type | | String | Header | Yes | Must be application/json;charset=UTF-8 |
| Output parameters, which are included in the uws response | | | | | |
| Parameter names | Type | | Location | Must return | Instructions |
| retCode | String | | Body | Yes | Return code (where 00000 means the request is successful, others represent errors, error codes and descriptions are listed in the Appendix Error Code Table) |
| retInfo | String | | Body | Yes | Return information for debugging, does not support internationalization, and cannot be directly displayed on the UI. |

* 1. Signature authentication
     1. Instructions

The caller needs to sign the request sent to uws, and the signature value of the signature calculation needs to be assigned to the sign attribute in the Header header (see the public section) for the server to perform signature verification.

* + 1. Introduction to parameters

**The string to be signed is**: url + Body +appId+appKey +timestamp；

**url** ：Refers to the path part of the requested interface address after removing https://domain name+port;

**Body**：Refers to the JSON string after the body part of the application sends the request to remove all whitespace characters. If there is no body, it is an empty string (not null);

**appId**：Properties in Header headers (see the public section description);

**appKey**：The appKey applied to the application on the Haigeek network cannot be sent in clear text;

**timestamp**：Properties in Header headers (see the public section description);

* + 1. Algorithm

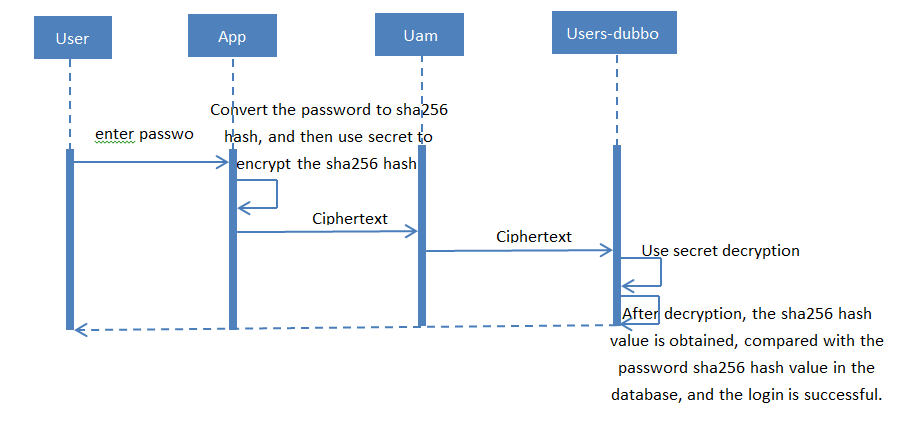
The signature algorithm is to calculate the 32-bit lowercase SHA-256 value for the signature string. See the appendix for an example of the algorithm.

* 1. Security of user password
     1. Safety instructions

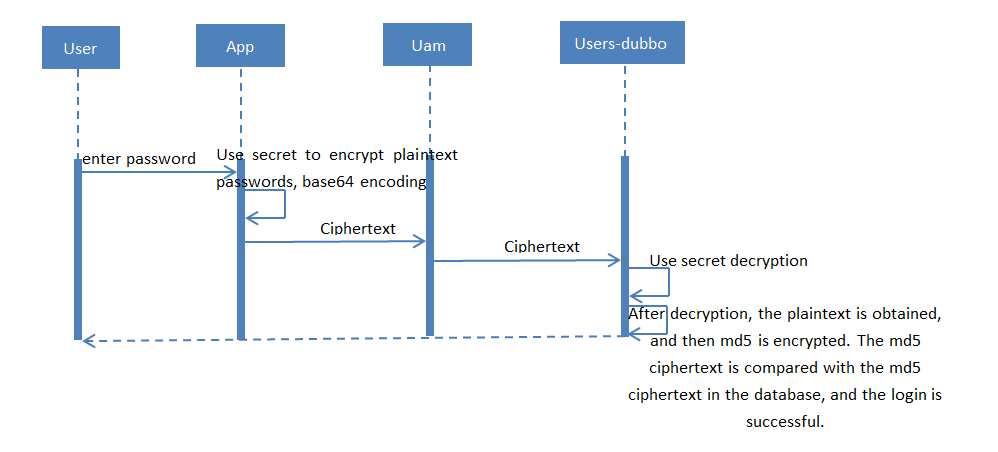
Since openapi USES md5+salt password processing mode, if the uws-uam interface also USES md5 mode, the app needs to transmit the password in clear text, which will reduce the security of the entire original gea environment, so the app transmission password must be encrypted. The login is encrypted as sha256, and the password is encrypted as sha256 when registering. Aes is used for encryption and base64 encoding.

* + 1. Password flow

North American environment:



European environment:



* + 1. Cipher encryption and decryption algorithm

algorithm：aes

secret key：App encryption and server decryption use the same secret key

algorithm code：See appendix 7.4

* 1. Internationalization

The retCode and retInfo returned by the interface response are not internationalized and are processed by the interface caller.

The internationalization of the interface involving business data is defined by passing the language parameter in the header. The specific international language code is shown in the appendix.

* 1. Data type qualification
     1. Field type description

|  |  |  |  |
| --- | --- | --- | --- |
| Qualified type | instructions | format | Json example |
| DateTime | Date time type string | yyyy-MM-dd hh:mm:ss | {“lgTime”:“2013-10-08 08:00:00”} |
| Date | Date type string | yyyy-MM-dd | {“lgDate”:“2013-10-08”} |
| String | String |  | {“address”:“street 123”} |
| int | int |  | {“age”:1234} |
| long | long |  | {“oid”:1234567890123} |
| double | double |  | {“price”:12.35} |
| boolean | boolean（true or false） |  | {“idOld”:true}。 |

* + 1. Null value description

To avoid parsing errors, the return parameters of each interface of uws do not return a null value.

Required parameters, whether input or output, must have a value and cannot be null.

Non-required parameters are as follows:

Numerical type data（int、long、double）Only numbers are returned, including positive numbers, zeros, and negative numbers.

Boolean type data（boolean）Only return true or false

The above basic types do not themselves contain null values.

DateTime, Date, String, and structure type data. If it is null, the corresponding attribute will not be returned.

E.g:

***DateTime type***

birthday is a DateTime type, not null:

{"name":"Tom","age":23,"birthday":"2013-10-08 08:00:00","address":{"city":"beijing","street":"haidian"} }

When birthday is null, the birthday attribute does not return:

{"name":"Tom","age":23, "address":{"city":"beijing","street":"haidian"} }

***String type***

Name is a String type，not null：

{"name":"Tom","age":23,"birthday":"2013-10-08 08:00:00","address":{"city":"beijing","street":"haidian"} }

When name is null, the name attribute is not returned:

{"age":23,"birthday":"2013-10-08 08:00:00","address":{"city":"beijing","street":"haidian"} }

***Structure type***

address is a struct type, not null:

{"name":"Tom","age":23,"birthday":"2013-10-08 08:00:00","address":{"city":"beijing","street":"haidian"} }

When address is null, the address attribute does not return:

{"name":"Tom","age":23,"birthday":"2013-10-08 08:00:00" }

3 User interface

3.1User registration

3.1.1 Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | User registration | | | | | | |
| Access address | /uam/v1/security/register | | | | | HTTP Method | POST |
| Functional description | Register new users | | | | | | |
| Token authentication | no (header can not be passed into accesstoken) | | | | | | |
| **Input parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| loginId | | String | Body | Yes | Mailbox, need to conform to mailbox format  Use the following regular expressions:  ^\w+([.+-]\w+)\*@\w+([.-]\w+)\*(\.\w{2,5})+$ | | |
| password | | String | Body | Yes | Length: between 6 and 16 characters  Must contain a combination of at least two of the following characters:  - at least one lowercase letter;  - at least one capital letter;  - at least one number;  - at least one special character：`~!@#$%^&\*()-\_=+\|[{}];:'",<.>/? and spaces  The password cannot be the same as the account number or the reverse writing of the account number;  Note: password strength is required by app. The incoming service is ciphertext. The secret key of aes = aes (sha256 (password)) is not provided in this document | | |
| captcha | | String | Body | Yes | Captcha, 4-bit alphanumeric combination. | | |
| userProfile | | Map | Body | No | Added to meet the user information needs of different applications. When the application needs to extend user attributes, it can apply to the cloud platform user system. The attributes that need to be extended are listed in the application, and the key, type and length corresponding to each attribute are also listed. | | |
| **Output parameters** | | | | | | | |
| Output standard output parameters. | | | | | | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| D00012 | Account number already exists |  |
| D00015 | Verification code error |  |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uam/v1/security/register |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type: application/json  Body:  {  "loginId": "14759167292@qq.com",  "password": "Abcd!123456",  "userProfile": {  "updateTime": "20141115",  "tel": "0596",  "companyCode": 333,  "address": "china",  "email": "848421322@qq.com",  "QQ": "848421322",  "name": "test",  "realname": "test"  }  } |
| Request response | {  "retCode": "00000",  "retInfo": "成功"  } |

* 1. User login

3.2.1 Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | User login | | | | | | |
| Access address | /uam/v1/security/login | | | | | HTTP Method | POST |
| Functional description | The user logs in to get accessToken and openId | | | | | | |
| Token authentication | No | | | | | | |
| **Input parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| loginId | | String | Body | Yes | Mailbox, need to conform to mailbox format  Use the following regular expressions:  ^\w+([.+-]\w+)\*@\w+([.-]\w+)\*(\.\w{2,5})+$ | | |
| password | | String | Body | Yes | Length: between 6 and 16 characters  Must contain a combination of at least two of the following characters:  - at least one lowercase letter;  - at least one capital letter;  - at least one number;  - at least one special character：  `~!@#$%^&\*()-\_=+\|[{}];:'",<.>/? and spaces  The password cannot be the same as the account number or the reverse writing of the account number;  Note: The password strength is the app requirement, and the incoming server is the sha256 ciphertext. | | |
| captcha | | String | Body | No | Captcha, 4-bit alphanumeric combination. When the password is entered three times or more, force the user to enter the verification code, such as five times, and lock the account for 5 hours | | |
| **Output parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| accessToken | | String | Header | Yes | Security token | | |
| userId | | String | Body | Yes | user id | | |
| status | | String | Body | Yes | 0 : activation ;   1. : unactivation; | | |

3.2.2 Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| D00002 | Account or password error |  |
| D00009 | Logon failure exceeded limit, need to use authentication code login | After the login failed twice, start the verification code login |
| D00010 | Account locked | After 5 failed login attempts, lock the account |
| D00015 | Verification code error |  |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uam/v1/security/login |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type: application/json  Body:  {  "loginId": "14759167292@qq.com",  "password": " Abcd!123456"  } |
| Request response | header:  accessToken: TGT2SI3VVPHX630U2VWJRYV3K25MM0  {  "retCode": "00000",  "retInfo": "成功",  "userId ": "1234567",  "status":"0"  } |

* 1. Send activation
     1. Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Resend the activation message | | | | | | |
| Access address | /uam/v1/security/sendActiveMail | | | | | HTTP Method | POST |
| Token authentication | No | | | | |  |  |
| Functional description | After successful registration, if the user fails to receive the activation mail due to mail network and other reasons, or if the user receives the activation mail but does not execute the activation, causing the expiration of the activation mail, the user can send the activation mail again through this interface. The prerequisite for using this interface is that the user is registered but not activated. The effective time to activate mail is 120 minutes. No login is required for access. | | | | | | |
| **Input parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| loginId | | String | Body | Yes | Mailbox, need to conform to mailbox format  Use the following regular expressions:  ^\w+([.+-]\w+)\*@\w+([.-]\w+)\*(\.\w{2,5})+$ | | |
| **Output parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| D00011 | Account not activated |  |
| D00017 | Account does not exist |  |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uam/v1/user/sendActiveMail |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type: application/json  Body:  {  "loginId": "14759167292@qq.com"  } |
| Request response | {  "retCode": "00000",  "retInfo": "成功"  } |

* 1. Log out
     1. Interface definition

|  |  |  |  |
| --- | --- | --- | --- |
| Interface name | Log out | | |
| Access address | /uam/v1/security/logout | HTTP Method | POST |
| Functional description | Mobile APP users exit Haier U+ cloud platform interface | | |
| Token authentication | Yes | | |
| **Input parameters** | | | |
| There is no | | | |
| **Output parameters** | | | |
| Output standard output parameters. | | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| D00005 | The Token is not created by this application and does not pass Token validation. | Operation is successful |
| D00016 | You are logged out or not logged in |  |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uam/v1/security/logout |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type: application/json |
| Request response | {  "retCode": "00000",  "retInfo": "成功"  } |

* 1. Query user information
     1. Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Query user information | | | | | | |
| Access address | /uam/v1/users/get | | | | | HTTP Method | POST |
| Functional description | Get the user information according to the login token | | | | | | |
| Token authentication | yes | | | | | | |
| **Input parameters** | | | | | | | |
| There is no | | | | | | | |
| **Output parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| userProfile | | Map | Body | No | User extension information, including nicknames, avatars, etc. | | |

* + 1. Retrun code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| D00008 | Illegal user | AccessToken error |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uam/v1/users/get |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json |
| Request response | {  “retCode”: “00000”,  “retInfo”: “正确”,  “userProfile”: {  “nickName”: ,  “avatar”: ,  “phone”: ,  “updateTime”: “20141115”,  “status”: null,  “tel”: “0596”,  “applyTime”: null,  “idcard”: null,  “companyName”: null,  “type”: null,  “postcode”: null,  “legalPerson”: null,  “contacts”: null,  “companyCode”: 333,  “businessLicense”: null,  “address”: “china”,  “contactsPhone”: null,  “email”: “848421322@qq.com”,  “QQ”: “848421322”,  “name”: “test”,  “realname”: “test”,  “idcardPhoto”: null  }  } |

* 1. User information modification
     1. Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | user information modification | | | | | | |
| Access address | /uam/v1/users/update | | | | | HTTP Method | POST |
| Functional description | Modify the extended properties of the current logged in user according to the token of the logged in person | | | | | | |
| Token authentication | Yes | | | | | | |
| **Input parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| userProfile | | Map | Body | Yes | User extension information, including nicknames, avatars, etc | | |
| **Output parameters** | | | | | | | |
| Output standard output parameters. | | | | | | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| D00008 | Illegal user | AccessToken error |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uam/v1/users/update |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json  Body:  {  "userProfile": {  "nickName": ,  "avatar": ,  "updateTime": "20141115",  "tel": "0596",  "companyCode": 333,  "address": "china",  "email": "848421322@qq.com",  "QQ": "848421322",  "name": "test",  "realname": "test"  }  } |
| Request response | {  "retCode": "00000",  "retInfo": "成功"  } |

* 1. Apply for reset password
     1. Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Apply for reset password | | | | | | |
| Access address | /uam/v1/security/pwd/applyReset | | | | | HTTP Method | POST |
| Functional description | The user applies for a reset password, and the link to the reset password is sent to the user's mailbox. The user clicks the link to reset the password. | | | | | | |
| Token authentication | No | | | | | | |
| **Input parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| loginId | | String | body | 是 | Email address, need to conform to the mailbox format Use the following regular expressions:  ^\w+([.+-]\w+)\*@\w+([.-]\w+)\*(\.\w{2,5})+$ | | |
| **Output parameters** | | | | | | | |
| Output standard output parameters. | | | | | | | |

3.7.2 Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| D00011 | Account not activated |  |
| D00017 | Account does not exist |  |

3.7.3Request example

|  |  |
| --- | --- |
| Request address | /uam/v1/security/pwd/applyReset |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json  Body:  {  "loginId": "13679101193@qq.com"  } |
| Request response | {  "retCode": "00000",  "retInfo": "成功"  } |

* 1. Get the picture verification code
     1. Interface definition

|  |  |  |  |
| --- | --- | --- | --- |
| Interface name | Get the picture verification code | | |
| Access address | /uam/v1/security/captcha | HTTP Method | POST |
| Functional description | Get the picture verification code | | |
| Token authentication | No | | |
| **Input parameters** | | | |
| There is no | | | |
| **Output parameters** | | | |
| File stream，image/png  response.setContentType("image/png"); | | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| Common error code | | |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uam/v1/security/captcha |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json |
| Request response | HTTP/1.1 200 OK  Date: Fri, 21 Nov 2008 01:57:21 GMT  Connection: close  Accept-Ranges: bytes  Pragma: No-cache  PATH=/; DOMAIN=.rd139.com;  Content-Type: image/png  Content-Length: 1381 |

1. Device class interface
   1. Public structure
      1. DeviceInfo

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DeviceInfo | | DeviceInfo |
| Field name | Type | Instructions | Remarks |
| deviceName | String | Device name,  equivalent to alias name |  |
| deviceId | String | Device ID |  |
| wifiType | String | Device wifitype | typeId |
| deviceType | String | Equipment category | The 8 bit codes of large and small categories |
| totalPermission | AuthInfo | The combination of authority and authority information |  |
| permissions | Permission[] | Permission information |  |
| online | Boolean | Whether online |  |
| deviceVersion | DeviceVersion | Device version information |  |

* + 1. AuthInfo

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Permission content information, at least one of which is true | | AuthInfo |
| Field name | Type | Instructions | Remarks |
| view | Boolean | Whether has check right |  |
| set | Boolean | Have configuration permissions |  |
| control | Boolean | Do you have control |  |

* + 1. Permission

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Permission information | | Permission |
| Field name | Type | Instructions | Remarks |
| auth | AuthInfo | Access content |  |
| authType | String | Permission types | Home: home sharing, share: personal sharing, owner: device owner, server: permission to appserver |

* + 1. Location

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Location information | | Location |
| Field name | Type | Instructions | Remarks |
| longitude | Double | longitude |  |
| latitude | Double | dimension |  |
| cityCode | String | City code |  |

* + 1. BaseProperty

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Basic properties | | BaseProperty |
| Field name | Type | Instructions | Remarks |
| brand | String | equipment brand |  |
| model | String | Equipment model |  |
| others | Map<String,String> | Other attributes |  |

* + 1. DeviceVersion

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Device version information | | DeviceVersion |
| Field name | Type | Instructions | Remarks |
| deviceId | String | Device ID |  |
| modules | Set<Module> | Module information |  |
| wifiType | String | Wifi type |  |
| deviceType | String | Device type | typeId |
| basePropertiy | BaseProperty | Brand information | See the definition of BaseProperty |
| location | Location | location information |  |

* + 1. Module

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Module information | | Module |
| Field name | Type | Instructions | Remarks |
| moduleId | String | module ID |  |
| moduleType | String | Module type |  |
| moduleInfos | Map<String,String> | Module other information | the collection of attributes and values |

* 1. Query the user device list
     1. Interface definition

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | |  | | |  | | | |  |  | |  |  | |
| Interface name | | | Query the user device list | | | | | | | | | | | | |
| Access address | | | /uds/v1/protected/deviceinfos | | | | | | | HTTP Method | | | GET | | |
| Functional description | | | Gets a list of devices that users themselves bind to and share with others. The device can be obtained by filtering through authType=owner. | | | | | | | | | | | | |
| **Input parameters** | | | | | | | | | | | | | | | |
| There is no | | | | | | | | | | | | | | | |
| **Output parameters** | | | | | | | | | | | | | | | |
| **Parameter name** | | | | | **Type** | **Location** | | **Required** | **Description** | | | | | | |
| deviceinfos | | | | | DeviceInfo[] | Body | | Yes | Device information list | | | | | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| G20202 | The current user does not match the device | The device id does not match the user |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uds/v1/protected/deviceinfos |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json |
| Request response | {  "deviceinfos": [  {  "deviceName": "测试",  "deviceId": "100823",  "wifiType": "0000000000000000000000000",  "deviceType": "00000",  " totalPermission": {  "view": true,  "set": true,  "control": true  }，"permissions": [  {  "authType": "home",  "auth": {  "view": true,  "set": false,  "control": false  }  },  {  "authType": "share",  "auth": {  "view": true,  "set": false,  "control": false  }  },  {  "authType": "owner",  "auth": {  "view": true,  "set": true,  "control": true  }  }  ]，"online": false,  "deviceVersion": {  "deviceId ": "100823",  "modules": [  {  "moduleId": "1",  "moduleType": "1",  "moduleInfos": [  {  "key1": "value1"  },  {  "key2": "value2"  }  ]  },  {  "moduleId": "2",  "moduleType": "1",  "moduleInfos": [  {  "key1": "value1"  },  {  "key2": "value2"  }  ]  },  {  "moduleId": "1",  "moduleType": "1",  "moduleInfos": [  {  "key1": "value1"  },  {  "key2": "value2"  }  ]  }  ],  "wifiType": "00000",  "deviceType": "00000",  "baseProperty": {  "brand": "1",  "model": "1",  "others": [  {  "key1": "value1"  },  {  "key1": "value1"  }  ]  }，"location": {  "longitude": "120",  "latitude": "45",  "cityCode ": "010"  }  },  },  {  "deviceName": "测试",  "deviceId": "100823",  "wifiType": "0000000000000000000000000",  "deviceType": "00000",  " totalPermission": {  "view": true,  "set": true,  "control": true  }，"permissions": [  {  "authType": "home",  "auth": {  "view": true,  "set": false,  "control": false  }  },  {  "authType": "share",  "auth": {  "view": true,  "set": false,  "control": false  }  },  {  "authType": "owner",  "auth": {  "view": true,  "set": true,  "control": true  }  }  ],  "online": false,  "deviceVersion": {  }  },  {  "deviceName": "测试",  "deviceId": "100823",  "wifiType": "0000000000000000000000000",  "deviceType": "00000",  " totalPermission": {  "view": true,  "set": true,  "control": true  }，"permissions": [  {  "authType": "home",  "auth": {  "view": true,  "set": false,  "control": false  }  },  {  "authType": "share",  "auth": {  "view": true,  "set": false,  "control": false  }  },  {  "authType": "owner",  "auth": {  "view": true,  "set": true,  "control": true  }  }  ],  "online": false,  "deviceVersion": {  }  }  ],  "retCode": "00000",  "retInfo": "成功" |

* 1. Binding equipment
     1. Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Binding equipment | | | | | | |
| Access address | /uds/v1/protected/bindDevice | | | | | HTTP Method | POST |
| Functional description | Establish a binding relationship between the user and the device | | | | | | |
| **Input parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| deviceId | | String | Body | Yes | Device id | | |
| name | | String | Body | Yes | Device name | | |
| bindReqSn | | String | Body | Yes | A random number generated by uSDK | | |
| data | | String | Body | Yes | Bind encrypted data (get the encrypted string from the SDK) | | |
| **Output parameters** | | | | | | | |
| Output standard output parameters. | | | | | | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| G20202 | The current user does not match the device | The device id does not match the user |
| G20904 | The device has been bound | Repeat binding |
| G20908 | Equipment out of safety period | The equipment has passed the safe period |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uds/v1/protected/bindDevice |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json  Body:  {  "deviceId": "12346",  "name": "我的设备",  "data": "6Kej56CB5ZmoIOivt+i+k+WFpeimgei/m+ihjOe8lueggeaIluino"  } |
| Request response | {  "retCode": "00000",  "retInfo": "成功"  } |

* 1. Unbundling equipment
     1. Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Unbundling equipment | | | | | | |
| Access address | /uds/v1/protected/{deviceId}/unbindDevice | | | | | HTTP Method | POST |
| Functional description | Unbind the user to the device | | | | | | |
| **Input parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| deviceId | | String | Body | Yes | Device id | | |
| **Output parameters** | | | | | | | |
| Output standard output parameters. | | | | | | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| G20202 | The current user does not match the device | The device id does not match the user |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uds/v1/protected/{deviceId}/unbindDevice |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json |
| Request response | {  "retCode": "00000",  "retInfo": "成功"  } |

* 1. Modify device alias
     1. Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Modify device alias | | | | | | |
| Access address | /uds/v1/protected/{deviceId}/updateAliasName | | | | | HTTP Method | PUT |
| Functional description | Update device alias | | | | | | |
| **Input parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| deviceId | | String | url | Yes | Device id | | |
| aliasName | | String | Body | Yes | New alias for device, no more than 50 characters, no special characters. | | |
| **Output parameters** | | | | | | | |
| Output standard output parameters. | | | | | | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| G20202 | The current user does not match the device | The device id does not match the user |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uds/v1/protected/{deviceId}/updateAliasName |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json  Body:  {  "aliasName ": "我的设备"  } |
| Request response | {  "retCode": "00000",  "retInfo": "成功"  } |

* 1. Modify device location information
     1. Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Modify device location information | | | | | | |
| Access address | /uds/v1/protected/{deviceId}/updateLocation | | | | | HTTP Method | PUT |
| Functional description | Modify device latitude and longitude, city code information | | | | | | |
| **Input parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| deviceId | | String | url | Yes | Device id | | |
| loc | | Location | Body | Yes | location information | | |
| **Output parameters** | | | | | | | |
| Output standard output parameters. | | | | | | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| G20202 | The current user does not match the device | The device id does not match the user |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uds/v1/protected/{deviceId}/updateLocation |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json  Body:  {  "loc": {  "longitude ": "120",  " latitude ": "45",  " cityCode ": "010"  }  } |
| Request response | {  "retCode": "00000",  "retInfo": "成功"  } |

* 1. Get device alias
     1. Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Get device alias | | | | | | |
| Access address | /uds/v1/protected/{deviceId}/aliasName | | | | HTTP Method | | GET |
| Functional description | Query device alias | | | | | | |
| **Input parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | | **Description** | |
| deviceId | | String | url | Yes | | Device id | |
| **Output parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | | **Description** | |
| aliasName | | String | Body | Yes | | Equipment alias | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| G20202 | The current user does not match the device | The device id does not match the user |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uds/v1/protected/{deviceId}/aliasName |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json |
| Request response | {  "aliasName ": "我的设备",  "retCode": "00000",  "retInfo": "成功"  } |

* 1. Obtain device location information
     1. Interface definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Obtain device location information | | | | | | |
| Access address | /uds/v1/protected/{deviceId}/location | | | | | HTTP Method | GET |
| Functional description | Obtain device location information | | | | | | |
| **Input parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| deviceId | | String | url | Yes | Device id | | |
| **Output parameters** | | | | | | | |
| **Parameter name** | | **Type** | **Location** | **Required** | **Description** | | |
| loc | | Location | Body | Yes | location information | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| G20202 | The current user does not match the device | The device id does not match the user |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uds/v1/protected/{deviceId}/location |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json |
| Request response | {  "loc"：{  "longitude": "120",  "latitude": "45",  "cityCode": "010"  },  "retCode": "00000",  "retInfo": "成功"  } |

* 1. Obtain device details
     1. Interface definition

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Obtain device location information | | | | | | | | | |
| Access address | /uds/v1/protected/{deviceId}/deviceInfo | | | | | | | | HTTP Method | GET |
| Functional description | Obtain device details | | | | | | | | | |
| **Input parameters** | | | | | | | | | | |
| **Parameter name** | | **Type** | **Location** | | **Required** | | **Description** | | | |
| deviceId | | String | url | | Yes | | Device id | | | |
| **Output parameters** | | | | | | | | | | |
| **Parameter name** | | **Type** | | **Location** | | **Required** | | **Description** | | |
| deviceVersion | | DeviceVersion | | Body | | Yes | | device information | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| G20202 | The current user does not match the device | The device id does not match the user |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uds/v1/protected/{deviceId}/deviceInfo |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json |
| Request response | {  " deviceVersion": {  "deviceId ": "100823",  "modules": [  {  "moduleId": "1",  "moduleType": "1",  "moduleInfos": [  {  "key1": "value1"  },  {  "key2": "value2"  }  ]  },  {  "moduleId": "2",  "moduleType": "1",  "moduleInfos": [  {  "key1": "value1"  },  {  "key2": "value2"  }  ]  },  {  "moduleId": "1",  "moduleType": "1",  "moduleInfos": [  {  "key1": "value1"  },  {  "key2": "value2"  }  ]  }  ],  "wifiType": "00000",  "deviceType": "00000",  "baseProperty": {  "brand": "1",  "model": "1",  "others": [  {  "key1": "value1"  },  {  "key1": "value1"  }  ]  }，"location": {  "longitude": "120",  "latitude": "45",  "cityCode ": "010"  }  },  "retCode": "00000",  "retInfo": "成功"  } |

* 1. Inquire if the device is online
     1. Interface definition

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Inquire if the device is online | | | | | | | | | |
| Access address | /uds/v1/protected/{deviceId}/isOnline | | | | | | | | HTTP Method | GET |
| Functional description | Users can see if the device is online | | | | | | | | | |
| **Input parameters** | | | | | | | | | | |
| **Parameter name** | | **Type** | **Location** | | **Required** | | **Description** | | | |
| deviceId | | String | url | | Yes | | Device id | | | |
| **Output parameters** | | | | | | | | | | |
| **Parameter name** | | **Type** | | **Location** | | **Required** | | **Description** | | |
| isOnline | | String | | Body | | Yes | | State information | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| G20202 | The current user does not match the device | The device id does not match the user |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uds/v1/protected/{deviceId}/isOnline |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:application/json |
| Request response | {  "online":false,  "retCode":"00000",  "retInfo":"成功"  } |

1. Capability class interface
   1. App version query
      1. Interface definition

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | App version query | | | | | | | |
| Access address | /uas/v1/appVersion/getLatest | | | | | | HTTP Method | POST |
| Functional description |  | | | | | | | |
| **Input parameters** | | | | | | | | |
| **Parameter name** | | | **Type** | **Location** | **Required** | **Description** | | |
| appId | | | String | Header | Yes | Application of the appId | | |
| **Output parameters** | | | | | | | | |
| **Parameter name** | | **Type** | | **Location** | **Required** | **Description** | | |
| version | | String | | Body | Yes | Version number (format 2015110501) | | |
| versionName | | String | | Body | Yes | Version name, returned as an empty string | | |
| description | | String | | Body | Yes | Description, can be returned as an empty string | | |
| resId | | String | | Body | Yes | The resource number or url of the resource store can be returned as an empty string | | |
| status | | Integer | | Body | Yes | State of the app | | |
| force | | String | | Body | Yes | Whether or not mandatory | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| 00000 | successful | Operation is successful |
| B00001 | Lack of required parameters | AppId is empty |
| C00002 | Appserver has no access | Appserver has no access |
| C00006 | Product configuration information is empty | Product configuration information is empty |
| C00007 | AppKey is empty | The appkey is empty according to the appId |
| D00001 | Digital signature error | Digital signature error |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uas/v1/appVersion/getLatest |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type: application/json  Body  {  " appId ":"MB-UZHSH-0000"  } |
| Request response | {  "retCode": "00000",  "retInfo": "成功",  "data": {  "version": "20140911",  "versionName": "V01.10.15.09101",  "description": "V01.10.15.09101",  "resId": "/uam/v1/resource/enabling/uzhsh/100013957366155388.jpg ",  "status": 1,  "force": "true"  }  } |

* 1. Upload resource file
     1. Interface definition

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Interface name | Upload resource file | | | | | | | | |
| Access address | /uas/v1/resource/uploadFile | | | | | | | HTTP Method | POST |
| Functional description | Upload the resource file to the server. (note: to use this interface, you need to contact the competent administrator to authorize the upload of this APPId, and configure the file format and size of the uploaded resource. Otherwise, there is no error in returning the file configuration.) | | | | | | | | |
| **Input parameters** | | | | | | | | | |
| **Parameter name** | | **Type** | | **Location** | | **Required** | **Description** | | |
| file | | multipart/form-data | | Body | | Yes | Uploaded file | | |
| description | | String | | Body | | Yes | File description, within 255 characters | | |
| ownerType | | Integer | | Body | | Yes | Owner type: 0: user, 1: device,9: other | | |
| **Output parameters** | | | | | | | | | |
| **Parameter name** | | **Type** | **Location** | | **Required** | | **Description** | | |
| resourceInfo | | ResourceInfo | Body | | Yes | | Upload resource information | | |

* + 1. Return code

|  |  |  |
| --- | --- | --- |
| error code | describe | scenario |
| C00002 | Appserver has no access | Appserver has no access |
| C00004 | Insufficient operation permission | Size format error |
| C00006 | Product configuration information is empty | Product configuration information is empty |
| C00007 | AppKey is empty | The appkey is empty according to the appId |
| D00008 | Illegal user | AccessToken error |

* + 1. Request example

|  |  |
| --- | --- |
| Request address | /uas/v1/resource/uploadFile |
| User request | Header：  appId:MB-ABC-0000  appVersion:2015110401  clientId:356877020056553-08002700DC94  sequenceId:08002700DC94-15110519074300001  accessToken: TGT1OY0RUUAH5D242SB68E9WX0W930  sign:bd4495183b97e8133aeab2f1916fed41  timestamp:1446639090139  language:zh-cn  timezone:8  Content-type:multipart/form-data  Body  {  "description":"测试文件上传新接口",  "ownerType":0  } |
| Request response | {  "retCode": "00000",  "retInfo": "成功"，  "resourceInfo":{  "id":30121,  "createTime":"2016-09-22 16:09:14",  "description":"测试文件上传新接口",  "fileType":"png",  "ownerType":0,  "fileName":"table.png",  "systemId":"SV-UZHSH-0000",  "url":"/uam/v1/resource/enabling/uzhsh/100013957366155388.jpg",  "creator":"100013957366155388"  }  } |

1. Appendix
   1. Common Error Codes

|  |  |
| --- | --- |
| **Error code** | **Info** |
| A00001 | Service unavailable |
| A00002 | Network exception |
| A00003 | Access or operation timeout |
| A00004 | Internal system error |
| A00005 | Database access exception |
| A00006 | Unknown exception |
| A00007 | Mail service exception |
| A00008 | Mail delivery failed |
| A00009 | The number of emails sent exceeded |
| B00001 | Missing required parameters |
| B00002 | Parameter type error |
| B00003 | Parameter values are out of range or not enumerated |
| B00004 | The parameters do not meet the rule requirements |
| B00006 | Parameter length error |
| B00007 | The parameters do not match the interface definition |
| C00001 | AppId and appKey validation failed |
| C00002 | AppServer has no access authorization |
| C00003 | Insufficient access |
| C00004 | Insufficient operation permission |
| C00005 | Repeated requests |
| C00006 | Unknown device type |
| C00007 | The appId configuration information is empty |
| C00008 | The appKey is empty |
| D00001 | Sign signature error |
| D00003 | Token does not exist and is not authenticated by Token |
| D00004 | Token has expired and is not authenticated by Token. |
| D00005 | Token is not created from this application and is not authenticated by Token. |
| D00006 | Session invalidation |
| D00007 | Not internal users |
| D00008 | User illegal |

* 1. Internationalized language code list

|  |  |  |  |
| --- | --- | --- | --- |
| **Language code** | **English name** | **Chinese name** | **Whether to support** |
| af | Afrikaans - South Africa | 南非荷兰语 | no |
| ar-ae | Arabic(U.A.E.) | 阿拉伯语 - 阿拉伯联合酋长国 | no |
| ar-bh | Arabic(Bahrain) | 阿拉伯语 - 巴林 | no |
| ar-dz | Arabic(Algeria) | 阿拉伯语 - 阿尔及利亚 | no |
| ar-eg | Arabic(Egypt) | 阿拉伯语 - 埃及 | no |
| ar-iq | Arabic(Iraq) | 阿拉伯语 - 伊拉克 | no |
| ar-jo | Arabic(Jordan) | 阿拉伯语 - 约旦 | no |
| ar-kw | Arabic(Kuwait) | 阿拉伯语 - 科威特 | no |
| ar-lb | Arabic(Lebanon) | 阿拉伯语 - 黎巴嫩 | no |
| ar-ly | Arabic(Libya) | 阿拉伯语 - 利比亚 | no |
| ar-ma | Arabic(Morocco) | 阿拉伯语 - 摩洛哥 | no |
| ar-om | Arabic(Oman) | 阿拉伯语 - 阿曼 | no |
| ar-qa | Arabic(Qatar) | 阿拉伯语 - 卡塔尔 | no |
| ar-sa | Arabic(Saudi Arabia) | 阿拉伯语 - 沙特阿拉伯 | no |
| ar-sy | Arabic(Syria) | 阿拉伯语 - 叙利亚 | no |
| ar-tn | Arabic(Tunisia) | 阿拉伯语 - 突尼斯 | no |
| ar-ye | Arabic(Yemen) | 阿拉伯语 - 也门 | no |
| be | Belarusian | 白俄罗斯语 | no |
| bg | Bulgarian | 保加利亚语 | no |
| ca | Catalan | 加泰罗尼亚语 | no |
| cs | Czech | 捷克语 | no |
| da | Danish | 丹麦语 | no |
| de | German(Standard) | 德语 - 标准 | no |
| de-at | German(Austrian) | 德语 - 奥地利 | no |
| de-ch | German(Swiss) | 德语 - 瑞士 | no |
| de-li | German(Liechtenstein) | 德语 - 列支敦士登 | no |
| de-lu | German(Luxembourg) | 德语 - 卢森堡 | no |
| el | Greek | 希腊语 | no |
| en | English | 英语 | Yes |
| en-au | English(Australian) | 英语 - 澳大利亚 | no |
| en-bz | English(Belize) | 英语 - 伯利兹 | no |
| en-ca | English(Canadian) | 英语 - 加拿大 | no |
| en-gb | English(British) | 英语 - 英国 | no |
| en-ie | English(Ireland) | 英语 - 爱尔兰 | no |
| en-jm | English(Jamaica) | 英语 - 牙买加 | no |
| en-nz | English(New Zealand) | 英语 - 新西兰 | no |
| en-tt | English(Trinidad) | 英语 - 特立尼达岛 | no |
| en-us | English(United States) | 英语 - 美国 | no |
| en-za | English(South Africa) | 英语 - 南非 | no |
| es | Spanish(Spain - Modern Sort) | 西班牙语 - 标准 | no |
| es-ar | Spanish(Argentina) | 西班牙语 - 阿根廷 | 否 |
| es-bo | Spanish(Bolivia) | 西班牙语 - 玻利维亚 | 否 |
| es-cl | Spanish(Chile) | 西班牙语 - 智利 | 否 |
| es-co | Spanish(Colombia) | 西班牙语 - 哥伦比亚 | 否 |
| es-cr | Spanish(Costa Rica) | 西班牙语 - 哥斯达黎加 | 否 |
| es-do | Spanish(Dominican Republic) | 西班牙语 - 多米尼加共和国 | 否 |
| es-ec | Spanish(Ecuador) | 西班牙语 - 厄瓜多尔 | 否 |
| es-gt | Spanish(Guatemala) | 西班牙语 - 危地马拉 | 否 |
| es-hn | Spanish(Honduras) | 西班牙语 - 洪都拉斯 | 否 |
| es-mx | Spanish(Mexican) | 西班牙语 - 墨西哥 | 否 |
| es-ni | Spanish(Nicaragua) | 西班牙语 - 尼加拉瓜 | 否 |
| es-pa | Spanish(Panama) | 西班牙语 - 巴拿马 | 否 |
| es-pe | Spanish(Peru) | 西班牙语 - 秘鲁 | 否 |
| es-pr | Spanish(Puerto Rico) | 西班牙语 - 波多黎各 | 否 |
| es-py | Spanish(Paraguay) | 西班牙语 - 巴拉圭 | 否 |
| es-sv | Spanish(El Salvador) | 西班牙语 - 萨尔瓦多 | 否 |
| es-uy | Spanish(Uruguay) | 西班牙语 - 乌拉圭 | 否 |
| es-ve | Spanish(Venezuela) | 西班牙语 - 委内瑞拉 | 否 |
| et | Estonian | 爱沙尼亚语 | 否 |
| eu | Basque | 巴斯克语 | 否 |
| fa | Farsi | 波斯语 | 否 |
| fi | Finnish | 芬兰语 | 否 |
| fo | Faeroese | 法罗语 | 否 |
| fr | French(Standard) | 法语 - 标准 | 否 |
| fr-be | French(Belgian) | 法语 - 比利时 | 否 |
| fr-ca | French(Canadian) | 法语 - 加拿大 | 否 |
| fr-ch | French(Swiss) | 法语 - 瑞士 | 否 |
| fr-lu | French(Luxembourg) | 法语 - 卢森堡 | 否 |
| gd | Gaelic(Scots) | 盖尔语 - 苏格兰 | 否 |
| gd-ie | Gaelic (Ireland) | 爱尔兰盖尔语 | 否 |
| he | Hebrew | 希伯来语 | 否 |
| hi | Hindi | 印地语 | 否 |
| hr | Croatian | 克罗地亚语 | 否 |
| hu | Hungarian | 匈牙利语 | 否 |
| in | Indonesian | 印度尼西亚语 | 否 |
| is | Icelandic | 冰岛语 | 否 |
| it | Italian(Standard) | 意大利语 - 标准 | 是 |
| it-ch | Italian(Swiss) | 意大利语 - 瑞士 | 否 |
| ja | Japanese | 日语 | 否 |
| ji | Yiddish | 意第绪语 | 否 |
| ko | Korean | 朝鲜语 | 否 |
| lt | Lithuanian | 立陶宛语 | 否 |
| lv | Latvian | 拉脱维亚语 | 否 |
| mk | Macedonian | 马其顿语 | 否 |
| ms | Malaysian | 马来语 - 马来西亚 | 否 |
| mt | Maltese | 马耳他语 | 否 |
| nl | Dutch(Standard) | 荷兰语 | 否 |
| nl-be | Dutch(Belgian) | 荷兰语 - 比利时 | 否 |
| no | Norwegian(Bokmal) | 挪威语 - 博克马尔 | 否 |
| pl | Polish | 波兰语 | 否 |
| pt | Portuguese(Standard) | 葡萄牙语 - 标准 | 否 |
| pt-br | Portuguese(Brazilian) | 葡萄牙语 - 巴西 | 否 |
| rm | Rhaeto-Romanic | 拉托-罗马语 | 否 |
| ro | Romanian | 罗马尼亚语 | 否 |
| ro-mo | Romanian(Moldavia) | 罗马尼亚语 - 摩尔多瓦 | 否 |
| ru | Russian | 俄语 | 是 |
| ru-mo | Russian(Moldavia) | 俄语 - 摩尔多瓦 | 否 |
| sb | Sorbian | 索布语 | 否 |
| sk | Slovak | 斯洛伐克语 | 否 |
| sl | Slovenian | 斯洛文尼亚语 | 否 |
| sq | Albanian | 阿尔巴尼亚语 | 否 |
| sr | Serbian(Cyrillic) | 塞尔维亚语 - 塞瑞利克 | 否 |
| sv | Swedish | 瑞典语 | 否 |
| sv-fi | Swedish(Finland) | 瑞典语 - 芬兰 | 否 |
| sx | Sutu | 苏图语 | 否 |
| sz | Sami (Lappish) | 芬兰地方语言 | 否 |
| th | Thai | 泰语 | 否 |
| tn | Tswana | 茨瓦纳语 | 否 |
| tr | Turkish | 土耳其语 | 否 |
| ts | Tsonga | 汤加语 | 否 |
| uk | Ukrainian | 乌克兰语 | 否 |
| ur | Urdu | 乌尔都语 - 巴基斯坦 | 否 |
| ve | Venda | 文达语 | 否 |
| vi | Vietnamese | 越南语 | 否 |
| xh | Xhosa | 科萨语 | 否 |
| zh-cn | Chinese(PRC) | 中文 - 中华人民共和国 | 否 |
| zh-hk | Chinese(Hong Kong SAR, PRC) | 中文 - 中华人民共和国香港特别行政区 | 否 |
| zh-sg | Chinese(Singapore) | 中文 - 新加坡 | 否 |
| zh-tw | Chinese(Taiwan Region) | 中文 - 台湾地区 | 否 |
| zu | Zulu | 祖鲁语 | 否 |

* 1. Example signature algorithm

String getSign(String appId, String appKey, String timestamp, String body,String url){：

URL urlObj = new URL(url);

url=urlObj.getPath();

appKey = appKey.trim();

appKey = appKey.replaceAll("\"", "");

if (body != null) {

body = body.trim();

}

if (!body.equals("")) {

body = body.replaceAll("", "");

body = body.replaceAll("\t", "");

body = body.replaceAll("\r", "");

body = body.replaceAll("\n", "");

}

log.info("body:"+body);

StringBuffer sb = new StringBuffer();

sb.append(url).append(body).append(appId).append(appKey).append(timestamp);

MessageDigest md = null;

byte[] bytes = null;

try {

md = MessageDigest.getInstance("SHA-256");

bytes = md.digest(sb.toString().getBytes("utf-8"));

} catch (Exception e) {

e.printStackTrace();

}

return BinaryToHexString(bytes);

}

String BinaryToHexString(byte[] bytes) {

StringBuilder hex = new StringBuilder();

String hexStr = "0123456789abcdef";

for (int i = 0; i < bytes.length; i++) {

hex.append(String.valueOf(hexStr.charAt((bytes[i] & 0xF0) >> 4)));

hex.append(String.valueOf(hexStr.charAt(bytes[i] & 0x0F)));

}

return hex.toString();

}

* 1. AES encryption and decryption algorithm code example

import javax.crypto.Cipher;

import javax.crypto.spec.IvParameterSpec;

import javax.crypto.spec.SecretKeySpec;

public class AESUtil {

private final static String CIPHER\_ALGORITHM = "AES";

private final static String CIPHER\_ALGORITHM\_FULL = "AES/CBC/PKCS5Padding";

private final static String VIPARA = "1269571569321021";

public static String encryptContent(String secret, String content) throws Exception {

IvParameterSpec zeroIv = new IvParameterSpec(VIPARA.getBytes());

SecretKeySpec key = new SecretKeySpec(secret.getBytes(), CIPHER\_ALGORITHM);

Cipher cipher = Cipher.getInstance(CIPHER\_ALGORITHM\_FULL);

cipher.init(Cipher.ENCRYPT\_MODE, key, zeroIv);

byte[] byteContent =content.getBytes("UTF-8");

byte[] result = cipher.doFinal(byteContent);

return new String(Base64Util.encode(result));

}

public static String decryptContent(String secret, String content) throws Exception {

byte[] byteMi = Base64Util.decode(content);

IvParameterSpec zeroIv = new IvParameterSpec(VIPARA.getBytes());

SecretKeySpec key = new SecretKeySpec(secret.getBytes(), CIPHER\_ALGORITHM);

Cipher cipher = Cipher.getInstance(CIPHER\_ALGORITHM\_FULL);

cipher.init(Cipher.DECRYPT\_MODE, key, zeroIv);

byte[] decryptedData = cipher.doFinal(byteMi);

return new String(decryptedData, "UTF-8");

}

}