

# USER MANUAL

## ZKBio CVSecurity 3rd Party API

---

Version: 1.2

Date: November, 2025

Software Version: ZKBio CVSecurity 6.0.0 or above Version

## Important Statements

Firstly, thank you for choosing our product. Before using the product, please read this manual carefully to avoid any unnecessary damage. Proper operation of the product will result in better performance and faster verification.

None of the content of this document shall be copied or delivered in any forms or by any means without the prior written consent of our company.

The product described in the manual may include the software whose copyrights are shared by the licensors, including our company. No one shall copy, distribute, revise, modify, extract, decompile, disassemble, decrypt, reverse engineering, lease, transfer, sub-license the software, or perform other acts of copyright infringement, unless such restrictions are prohibited by applicable laws or such actions are approved by respective copyright holders.



Information provided in this manual may differ from actual technical specifications due to the constant development of products. Our company claims no responsibility for any disputes arising out of any discrepancy between actual technical parameters and those described in this document. The document is subject to change without prior notice.

# Table of Contents

1 Interface Description .....	4
1.1 Authorize API Level .....	4
1.2 Interface Parameter .....	5
2 Interface Details .....	6
2.1 Personnel .....	6
2.1.1 Person Interface .....	6
2.1.2 Department Interface .....	26
2.1.3 Card Interface .....	31
2.1.4 Person BioTemplate Interface .....	33
2.2 Access Control .....	38
2.2.1 Access Control Device Interface .....	38
2.2.2 Door Interface .....	41
2.2.3 Reader Interface .....	51
2.2.4 Access Control Level Interface .....	54
2.2.5 Access Control transaction Interface .....	63
2.2.6 Advanced Access Control Interface .....	102
2.3 Time & Attendance .....	104
2.3.1 Person Interface of Attendance Area .....	104
2.3.2 Edit Attendance Personnel Information .....	108
2.3.3 Att Device Interface .....	110
2.3.4 Apply for Abnormal Attendance .....	111
2.3.5 Att Transaction Interface .....	117
2.4 Offline Consumption .....	121
2.4.1 Offline Consumption and Consumption Record Interface .....	121
2.5 Online Consumption .....	124
2.5.1 Online Consumption Account Interface .....	124
2.5.2 Online Consumption and Consumption Record Interface .....	126
2.6 Elevator Control .....	130

2.6.1 Elevator Device Interface .....	130
2.6.2 Elevator Control Level Interface .....	134
2.6.3 Elevator Transaction Interface .....	138
2.6.4 Floor Interface .....	143
2.7 Visitor Management .....	147
2.7.1 Visitor Registration Check Out Interface .....	147
2.7.2 Visitor Reservation Interface .....	155
2.7.3 Visitor Level Interface .....	157
2.8 Parking .....	157
2.8.1 Parking Basic Management Interface .....	157
2.8.2 Park Authorization Interface .....	167
2.8.3 Park Transaction Interface .....	174
2.8.4 Parking Cost Interface .....	182
2.9 Entrance Control .....	185
2.9.1 Entrance Control Device Interface .....	185
2.9.2 Entrance Control Gate Interface .....	188
2.9.3 Entrance Control Gate Interface .....	195
2.9.4 Entrance Control Gate Reader Interface .....	203
2.9.5 Entrance Control Device transaction Interface .....	206
2.10 Facekiosk .....	225
2.10.1 FaceKiosk Personal Information .....	225
2.10.2 FaceKiosk Area Personal .....	225
2.10.3 FaceKiosk Device Interface .....	229
2.10.4 Media Interface .....	232
2.11 Smart Video Surveillance .....	234
2.11.1 Smart Video Device Interface .....	234
2.12 Intrusion Alarm .....	243
2.12.1 Intrusion Event Record Interface .....	243
2.13 Space Management .....	246

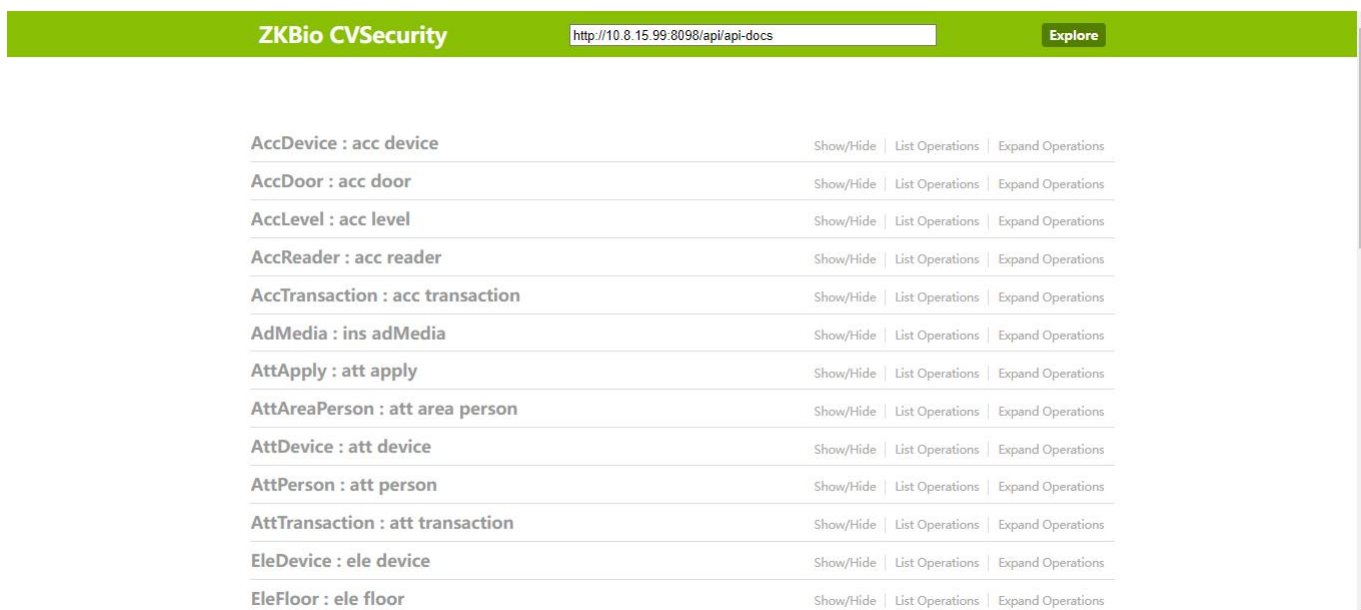
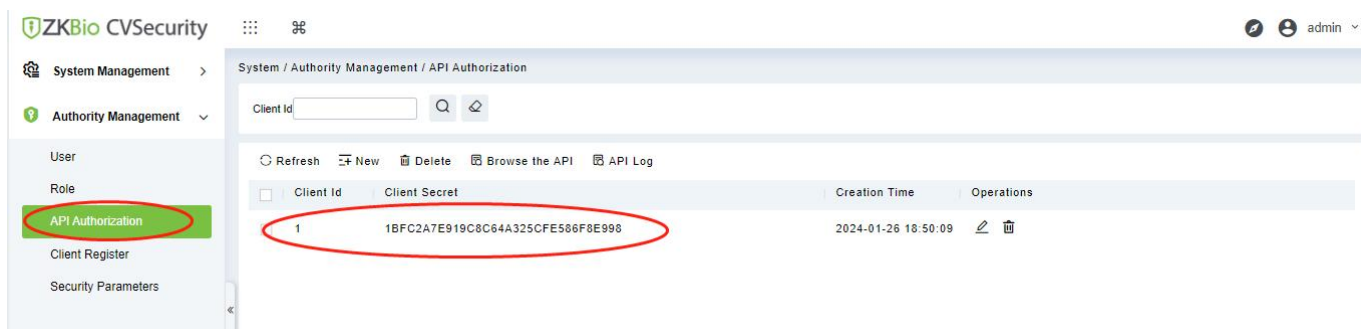
2.13.1 Space Device Interface .....	246
2.13.2 Space Interface .....	249
2.13.3 Space Facility Interface .....	257
2.13.4 Space Service Interface .....	262
2.13.5 Space Reservation Interface .....	267
3 Appendix .....	278
3.1 Error Code .....	278

# 1 Interface Description

**ZKBio CVSecurity API** is dedicated to the platform data connection for the third-party system. The third-party system can read and set business data flexibly, which effectively reduces the complexity of the third-party system business integration and provides convenient, fast standard connection mode and data structure.

## 1.1 Authorize API Level

The API Authorization menu will only be displayed after the API license is activated. Open the corresponding authorized API menu and add the corresponding API client information as shown below:



Based on the standard http interface definition style, in consistent with RESTFUL API style, supports https. Parameter descriptions are as following:

serverIP: ZKBio CVSecurity server or computer IP, such as: 192.168.1.100;

serverPort: The server port of ZKBio CVSecurity, such as: 8088;

access\_token: API access token is to check whether the requested permission is allowed or denied. All request should pass through a valid token, such as: apitoken;

## 1.2 Interface Parameter

<b>General Response Result</b>	<pre>{   "code": 0,   "message": "string",   "data": {} }</pre>
<b>General Response Result Parameter Description</b>	<p>code: error code, less than 0 means failure, more than 0 means success;  msg: Error message  data: can refer to each interface description</p>
<b>Basic Parameter Description</b>	<p>Sn: device serial number  Pin: pin  pageSize: page size;  pageNo: page no;</p>

## 2 Interface Details

### 2.1 Personnel

#### 2.1.1 Person Interface

##### 2.1.1.1 Add/ Edit Person Info [person/add]

<b>Post Request URL</b>	http://serverIP:serverPort/api/person/add?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "pin": "1234",   "deptCode": "1",   "name": "name",   "lastName": "person name (English)",   "gender": "F",   "cardNo": "card no",   "personPhoto": "picture-base64",   "accLevelIds": "access level group ids",   "accStartTime": "the start time of person valid access ",   "accEndTime": "the end time of person valid access " }</pre>
<b>Request Parameter Description</b>	<p>Pin: pin, required</p> <p>deptCode: department code;</p> <p>name: name;</p> <p>lastName: person name (English)</p> <p>gender: gender, male, female;</p> <p>cardNo: card no;</p> <p>personPhoto: personnel comparison picture. "picture turn into base64 format"</p> <p>accLevelIds: access level groupid List, multiple comma separated, string; such as :</p>

	402856aa6c3c5063016c3cb3a0360005,402856aa6bff4b1a016bff5374800c0e,402856aa6c3baabb016c3bb37365000a; accStartTime: access start time, such as :2017-08-04 15:45:00; accEndTime: access end time, such as :2017-08-04 23:59:59;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.1.1.2 Delete Person Info by Pin [person/delete/{pin}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/person/delete/{pin}?access_token={apitoken}
<b>Request Mode</b>	DELETE
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pin: pin, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.1.1.3 Delete Person Info by Pin (With Total Number) [person/delete/{pin}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/person/delete?pin={pin}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pin: pin, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result</b>	Refer to General Response Result Parameter Description <a href="#">Interface</a>

Description	<a href="#">Parameter</a>
-------------	---------------------------

### 2.1.1.4 Get Person Info by Pin [person/get/{pin}]Get person Info by Pin [person/get/{pin}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/person/get/{pin}?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pin: pin, required;
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "pin": "pin",     "deptCode": department code,     "deptName": "department name",     "name": "name",     "lastName": "person name (English)",     "gender": "gender",     "cardNo": "card no"   } } </pre>
<b>Response Result Description</b>	data: person data info; pin: pin; gender: gender, M: male, F: female; deptCode: department code; deptName: department name; name: name; lastName: person name (English); cardNo: card no;

### 2.1.1.5 Get Person Info by Pin (With Total Number) [person/get/{pin}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/person/get?pin={pin}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pin: pin, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "pin": "pin",     "deptCode": department code,     "deptName": "department name",     "name": "name",     "lastName": "person name (English)",     "gender": "gender",     "cardNo": "card no"     "vislightPhoto":base64 photo,     "vislightPhotoPath": "/upload/pers/user/cropface/789789/789789.jpg"   } }</pre>
<b>Response Result Description</b>	<p>data: person data info;  pin: personnel ID;  gender: gender, M: male, F: female;  deptCode: department code;  deptName: department name;  name: name;</p>

	<p>lastName: person name (English);</p> <p>cardNo: card no;</p> <p>vislightPhoto: Comparison photo (base64 encoded);</p> <p>vislightPhotoPath: Path to the comparison photo.</p>
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.1.1.6 Get Personnel Information according to the PIN Array and the Department Code Array

<b>Post Request URL</b>	http://serverIP:serverPort/api/person/getPersonList?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "pins": "1,2,3",   "deptCodes": "1,2",   "pageNo":1,   "pageSize":10 }</pre>
<b>Request Parameter Description</b>	<p>pageNo: page number.(Page number is necessary and must be greater than zero.)</p> <p>pageSize: Number of entries per page is necessary and must be greater than zero.;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "pin": "PIN Personal identification number",</pre>

	<pre> "deptCode": "Department Code", "deptName": "Department Name", "name": "First Name", "lastName": "Last Name", "gender": "Gender", "cardNo": Card number, "personPhoto": Personal photo( base64 photo), "vislightPhoto":Facial recognition photo(base64 photo), "vislightPhotoPath": "/upload/pers/user/cropface/789789/789789.jpg" } } </pre>
<p><b>Response Result Description</b></p>	<p>data: Personnel information date;</p> <p>pin: Personal identification number;</p> <p>gender: gender, M is male, F is female;</p> <p>deptCode: Department Code;</p> <p>deptName: Department Name;</p> <p>name: First Name;</p> <p>lastName: Last Name;</p> <p>cardNo: Card number;</p> <p>personPhoto: personal photo (base64 photo);</p> <p>vislightPhoto: Facial recognition photo (base64 photo);</p> <p>vislightPhotoPath: Facial recognition photo path;</p>

### 2.1.1.7 Get Dynamic QR Code by Personal Identification Number

<b>Post Request URL</b>	http://serverIP:serverPort/api/person/getQrCode/{pin}?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "pin": "123" }</pre> <p>http://serverIP:serverPort/api/person/getQrCode/{pin}?access_token={apitoken}</p>
<b>Request Parameter Description</b>	pin: Personal identification number and it is necessary
<b>Response Result</b>	<pre>{   "code":0,   "message": "success",   "data":   "2#6SQLlaSLhprGhpkCgFrhHfdyZXFJxK2DLy+oLVolmol=" }</pre>
<b>Response Result Description</b>	data: dynamic QR code (base64 data);

### 2.1.1.8 Get Dynamic QR Code by Personal Identification Number--V2

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/person/getQrCode?pin={pin}&access_token={apitoken}
<b>Request Mode</b>	POST

<b>Request Content</b>	{ "pin": "123" }
<b>Request Parameter Description</b>	pin: Personnel ID,required
<b>Response Result</b>	{ "code":0, "message": "success", "data": "2#6SQLlaSLhprGhpkCgFrhHfdyZXFJxK2DLy+oLVolmol=" }
<b>Response Result Description</b>	data: dynamic QR code (base64 data);

### 2.1.1.9 Resignation

<b>Post Request URL</b>	http://serverIP:serverPort/api/person/leave?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	{ "leaveDate": "2019-06-10", "leaveType": "1", "pin": "123" }
<b>Request Parameter Description</b>	leaveDate: Leave Date leaveType: Leave Type

	pin: Personal identification number and it is necessary
<b>Response Result</b>	<pre>{   "code":0,   "message": "success",   "data": null }</pre>
<b>Response Result Description</b>	http://serverIP:serverPort/api/person/leave?access_token={apitoken}

### 2.1.1.10 Add/Edit Personnel Basic Information

<b>Post Request URL</b>	http://serverIP:serverPort/api/person/addPersonnelBasicInfo?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "pin": "1234567",   "deptCode": "1",   "birthday": "2016-07-15",   "cardNo": "123456789",   "certNumber": "123456",   "certType": "2",   "email": "123@zkteco.com",   "gender": "F",   "hireDate": "2019-06-10",   "isDisabled": false, }</pre>

	<pre> "isSendMail": true,  "lastName": "lastName",  "mobilePhone": "15123456789",  "name": "max",  "personPwd": "123456",  "ssn": "111111",  "supplyCards": "987643",  } </pre>
<p><b>Request Parameter Description</b></p>	<p>pin: Personal identification number and it is necessary;</p> <p>deptCode: Department Code;</p> <p>name: First Name;</p> <p>lastName: Last Name;</p> <p>gender: Gender, M: male, F: female;</p> <p>cardNo: card number. Only cardNO is set as an empty string the card number be deleted.If it is not set, the original card number will be maintained. The format is: "cardno":</p> <p>birthday: Birthday;</p> <p>certNumber: Certificate number;</p> <p>certType: certificate type.1: Second generation ID card, 2: Passport, 3: Driver's License, 4: Work Permit, 5: Citizen Card, 6: Driver's License;1000: Xiamen Social Security Card, 1001: Hong Kong, Macao passport;</p> <p>email: Email address, such as 123@zkteco.com;</p> <p>hireDate: employee's entry date, such as: " 2019-06-10";</p>

	<p>isDisabled: whether it is a prohibited list. True: Yes; False: No</p> <p>isSendMail: whether to send mail. True: send; False: do not send;</p> <p>mobilePhone: phone number,such as: "15123456789",</p> <p>personPwd: device authentication password such as: "123456";</p> <p>ssn: social security number;</p> <p>supplyCards: Secondary cards. To use this function, you need to set up one person multiple cards in the personnel parameter setting. You can set multiple cards, which are separated by commas,such as: "987643, 2233452";</p>
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.1.1.11 Update Personnel Photo by PIN

<b>Post Request URL</b>	<p>http://serverIP:serverPort/api/person/updatePersonnelPhoto</p> <p>?access_token={apitoken}</p>
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "pin": "1234567",   "personPhoto": "string", }</pre>
<b>Request Parameter Description</b>	<p>pin: Personal identification number and it is necessary;</p> <p>personPhoto: Personnel photos and it is in Base64 format.If no value is transferred, no processing will be performed</p>

<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.1.1.12 Get Personnel Information according to the PIN Array and the Department Code Array (With Total Number)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/person/getPersonList?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "pins": "1,2,3",   "deptCodes": "1,2",   "pageNo":1,   "pageSize":10 }</pre>
<b>Request Parameter Description</b>	<p>pageNo: page number.(Page number is necessary and must be greater than zero.)</p> <p>pageSize: Number of entries per page is necessary and must be greater than zero.</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0, </pre>

```
"size": 1,  
"total": 1,  
"data": [{  
  "id": "8a888e238488391e018488e253ba016d",  
  "pin": "1",  
  "deptCode": "1",  
  "deptName": "department name",  
  "name": "seven",  
  "lastName": "",  
  "gender": "F",  
  "birthday": "2022-11-18",  
  "cardNo": "1111111",  
  "supplyCards": "",  
  "personPhoto": "",  
  "selfPwd": "e10adc3949ba59abbe56e057f20f883e",  
  "isSendMail": false,  
  "mobilePhone": "18094041582",  
  "personPwd": "",  
  "carPlate": null,  
  "email": "",  
  "ssn": null,  
  "accLevelIds": "8a888e2384847238018484748507065a",  
  "accStartTime": null,
```

	<pre> "accEndTime": null,  "certType": "",  "certNumber": "",  "photoPath": "",  "hireDate": null,  "isDisabled": false,  "vislightPhoto": "",  "vislightPhotoPath": "/upload/pers/user/cropface/1/1.jpg"      }},  "offset": 0,  "lastPage": true      } } </pre>
<p><b>Response Result Description</b></p>	<p>data: the result returned;</p> <p>page: The page number of the current request;</p> <p>size: The number of data requested by the current page number;</p> <p>total: total quantity;</p> <p>data: Personnel information date;</p> <p>pin: Personal identification number;</p> <p>gender: gender, M is male, F is female;</p> <p>deptCode: Department Code;</p> <p>deptName: Department Name;</p> <p>name: First Name;</p> <p>lastName: Last Name;</p> <p>cardNo: Card number;</p>

	<p>supplyCards: secondary cards;</p> <p>birthday: date of birth;</p> <p>selfPwd: self-service password;</p> <p>personPhoto: person's headshot base64 photo;</p> <p>vislightPhoto: comparison photo (base64 photo);</p> <p>vislightPhotoPath: path of the comparison photo;</p> <p>isSendMail: whether mailbox notification;</p> <p>mobilePhone: mobile phone number;</p> <p>personPwd: device authentication password;</p> <p>carPlate: car plate;</p> <p>email: email address;</p> <p>ssn : social security number;</p> <p>accLevelIds: access control privilege group id;</p> <p>accStartTime: access control privilege effective time from</p> <p>accEndTime: access control rights validity time Stop</p> <p>certType: type of document (2: ID card, 3: passport, 4: driver's licence, 5: work permit, 6: citizen card, 7: driving licence, 8: other, 1000: Xiamen social security card)</p> <p>certNumber: certificate number</p> <p>photoPath: photo path (base64)</p> <p>hireDate: entry time (2019-06-10)</p> <p>isDisabled: disabled list</p>
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.1.1.13 Reinstatement[api/person/reinstated]

<b>Post Request URL</b>	http://serverIP:serverPort/api/person/reinstated?access_token={api token}
<b>Request Mode</b>	POST
<b>Request Content</b>	{ "pin": "1234567",

	<pre> "deptCode": "1",  "birthday": "2016-07-15",  "carPlate": "987654",  "cardNo": "123456789",  "certNumber": "123456",  "certType": "2",  "email": "123@zkteco.com",  "gender": "F",  "hireDate": " 2019-06-10",  "isDisabled": false,  "isSendMail": true,  "lastName": "lastName",  "mobilePhone": "15123456789",  "name": "max",  "personPhoto": "string",  "personPwd": "123456",  "ssn": "111111",  "supplyCards": "987643",  "accEndTime": "2019-07-14 08:56:00",  "accLevelIds": "access level id",  "accStartTime": "2023-07-14 08:56:00"  } </pre>
<p><b>Request Parameter Description</b></p>	<p>pin: personnel number, required;</p>

deptCode: department number;

name: name;

lastName: the name of the person, only useful in English;

gender: gender, M male, F female;

cardNo: card number; only when set cardNo as an empty string will be deleted, do not set as to maintain the original card number, the format is: "cardNo": " ";

birthday: birthday date;

carPlate: car plate number; Multiple license plates are divided by English commas, such as "A12345,B12345"

When setting carPlate as an empty string, the license plate will be deleted, and it is not regarded as keeping the original license plate, with the format: "carPlate": "" ;

certNumber: the number of the certificate;

certType: certificate type, 2: second-generation ID card, 3: passport, 4: driver's licence, 5: work permit, 6: citizen card, 7: driver's licence; 1000: Xiamen social security card, 1001: Hong Kong and Macao Travel Permit;

email: email address, such as 123@zkteco.com;

hireDate: the date the person was hired, e.g., " 2019-06-10";

isDisabled: whether the list is banned, true: yes; false: no

isSendMail: whether to send mail, true: send; false: do not send;

mobilePhone: mobile phone number, e.g., "15123456789",

personPhoto: person photo, base64 format;

personPwd: device authentication password, e.g., "123456";

	<p>ssn: social security number;</p> <p>supplyCards: secondary cards, to use this feature, you need to enable multiple cards for one person in the personnel parameter settings, you can set more than one, more than one separated by commas, such as: "987643, 2233452";</p> <p>accLevelIds: access control rights group id collection, more than one with comma separated, such as: 402856aa6c3c3c5063016c3cb3a0360005,402856aa6bff4b1a016bff5374800c0e,402856aa6c3baabb016c3bb37365000a .;</p> <p>accStartTime: access control effective start time, e.g. 2017-08-04 15:45:00;</p> <p>accEndTime: access control valid end time, e.g., 2017-08-04 23:59:59;</p>
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

#### 2.1.1.14 Bulk Add/Edit Personnel Information [api/v2/person/addPersons]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/person/addPersons?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>[{   "pin": "1234567",   "deptCode": "1",   "birthday": "2016-07-15",</pre>

```
"carPlate": "987654",  
"cardNo": "123456789",  
"certNumber": "123456",  
"certType": "2",  
"email": "123@zkteco.com",  
"gender": "F",  
"hireDate": "2019-06-10",  
"isDisabled": false,  
"isSendMail": true,  
"lastName": "lastName",  
"mobilePhone": "15123456789",  
"name": "max",  
"personPhoto": "string",  
"personPwd": "123456",  
"ssn": "111111",  
"supplyCards": "987643",  
"accEndTime": "2019-07-14 08:56:00",  
"accLevelIds": "access level id",  
"accStartTime": "2018-07-14 08:56:00"  
},  
{  
"pin": "123",  
"deptCode": "1"
```

	}]
<b>Request Parameter Description</b>	<p>Pin: pin, required</p> <p>deptCode: department code;</p> <p>name: name;</p> <p>lastName: person name (English)</p> <p>gender: gender, male, female;</p> <p>cardNo: card no;</p> <p>personPhoto: personnel comparison picture. "picture turn into base64 format"</p> <p>accLevelIds: access level groupid List, multiple comma separated, string; such as : 402856aa6c3c5063016c3cb3a0360005,402856aa6bff4b1a016bff5374800c0e,402856aa6c3baabb016c3bb37365000a;</p> <p>accStartTime: access start time, such as :2017-08-04 15:45:00;</p> <p>accEndTime: access end time, such as :2017-08-04 23:59:59;</p>
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.1.1.15 Bulk Delete Personnel Information by PIN [api/v2/person/deleteByPins]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/person/deleteByPins?access_token={apitoken}&pins={pin1,pin2}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pins: personnel id1, personnel id 2 required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result</b>	Refer to General Response Result Parameter Description <a href="#">Interface</a>

Description	<a href="#">Parameter</a>
-------------	---------------------------

### 2.1.1.16 Detect whether a Facial Photo can be Extracted or Not

[api/v2/person/detectFace]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/person/detectFace ?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "personPhoto": "base64 data", }</pre>
<b>Request Parameter Description</b>	personPhoto: required, user photo, base64 format;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

## 2.1.2 Department Interface

### 2.1.2.1 Add /Edit Department[department/add]

<b>Post Request URL</b>	http://serverIP:serverPort/api/department/add?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "name": "department name",   "code": "456", }</pre>

	"parentCode": "123" }
<b>Request Parameter Description</b>	Name: department name, required; code: department code, required; parentCode: parent department number;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.1.2.2 Delete Department by Code [department/delete/{code}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/department/delete/{code}?access_token={apitoken}
<b>Request Mode</b>	DELETE
<b>Request Content</b>	None
<b>Request Parameter Description</b>	code: department code, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.1.2.3 Delete Department by Code[department/delete]--V2

<b>Post Request URL</b>	http://serverIP:serverPort/api/department/delete?code={code}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	code: department id, required;

<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

#### 2.1.2.4 Get Department Info by Code [department/get/{code}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/department/get/{code}?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Code: department code, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "name": "4444",     "code": "3333",     "parentCode": "2"   } }</pre>
<b>Response Result Description</b>	Data: department object Name: department name; Code: department code; parentCode: parent department code

#### 2.1.2.5 Get Department Info by Code [department/get/{code}]-V2

<b>Post Request URL</b>	http://serverIP:serverPort/api/department/get?code={code}&access_token={apitoken}
<b>Request Mode</b>	GET

<b>Request Content</b>	None
<b>Request Parameter Description</b>	Code: department code, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "name": "4444",     "code": "3333",     "parentCode": "2"   } }</pre>
<b>Response Result Description</b>	Data: department object Name: department name; Code: department code; parentCode: parent department code

### 2.1.2.6 Get Department List [department/get?code]

<b>Post Request URL</b>	http://serverIP:serverPort/api/department/getDepartmentList?pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: Page number, required, must be greater than 0. pageSize: Number of items per page, required, must be greater than 0.
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {</pre>

	<pre> "page": 0,  "size": 2,  "total": 2,  "data": [    {      "name": "department name",      "code": "1",      "sortNo": 1,      "parentCode": null    },    {      "name": "55",      "code": "2",      "sortNo": 99999,      "parentCode": "1"    }  ],  "offset": 0,  "lastPage": true  }  } </pre>
<p><b>Response Result Description</b></p>	<p>sortNo: Sorting number;  name: Department name;  code: Department code;</p>

parentCode: Parent department code.

## 2.1.3 Card Interface

### 2.1.3.1 Get Card List by Personnel ID [card/getCards/{pin}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/card/getCards/{pin}?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pin: Personnel ID, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "pin": "1234",       "cardNo": "111",       "cardType": "0"     }   ] }</pre>
<b>Response Result Description</b>	<p>data: card number List;</p> <p>pin: Personnel ID;</p> <p>cardNo: card no;</p> <p>cardType: card type, 0 : master card, 1: sub card, a person only uses one card when the system does not open the function of one more card</p>

### 2.1.3.2 Get Card List by Personnel ID [card/getCards/{pin}]-V2

<b>Post Request URL</b>	http://serverIP:serverPort/api/card/getCards?pin={pin}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pin: Personnel ID, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "pin": "1234",       "cardNo": "111",       "cardType": "0"     }   ] }</pre>
<b>Response Result Description</b>	<p>data: card number List;</p> <p>pin: Personnel ID;</p> <p>cardNo: card no;</p> <p>cardType: card type, 0 : master card, 1: sub card, a person only uses one card when the system does not open the function of one more card</p>

### 2.1.3.3 Set Person Card Info[card/set]

<b>Post Request URL</b>	http://serverIP:serverPort/api/card/set?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	{

	<pre>"pin": "1234", "cardNo": "212121", "cardType": "0" }</pre>
<b>Request Parameter Description</b>	<p>Pin: pin, required</p> <p>cardNo: card number, required;</p> <p>cardType: card type, 0: master card, 1: sub card, the system does not open multi-card function, sub card setting invalid</p>
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

## 2.1.4 Person BioTemplate Interface

### 2.1.4.1 Add /Edit Personnel BioTemplate

<b>Post Request URL</b>	http://serverIP:serverPort/api/bioTemplate/add?access_token={apitoken}
<b>Request Mode</b>	Post
<b>Request Content</b>	<pre>{   "personId": "666",   "validType": "1",   "template": "fsfsfsfsfsfsfsfsfsfsfsfsfsfs",   "templateNo": "3" }</pre>
<b>Request Parameter Description</b>	<p>personId: personnel ID, required;</p> <p>validType: 1:common, 3:duress, required;</p> <p>Template: personnel fingerprint template, required;</p> <p>templateNo: finger number, required,range[0-9];</p>
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>

<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>
------------------------------------	--------------------------------------------------------------------------------------------

### 2.1.4.2 Retrieve BioTemplate Information by Personnel Number

<b>Post Request URL</b>	http://serverIP:serverPort/api/bioTemplate/getFgListByPin/{pin}?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	NONE
<b>Request Parameter Description</b>	Pin: personnel ID, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "40280bb470e118510170e13f57dc001a",     "pin": "3170156",     "validType": "1",     "bioType": 1,     "version": "10",     "template": "TWITUzlxAAAeKikECAUHCc7QAAAcK2kBAAA     AhNcowCrDALI P3.....(Biotemplate)",     "templateNo": "6",     "templateNoIndex": 0   } }</pre>
<b>Response Result</b>	data: Personnel fingerprint data information;

<b>Description</b>	<p>id: Personnel fingerprint ID;</p> <p>pin: Personnel number;</p> <p>validType: Validity status;</p> <p>bioType: Biometric template type;</p> <p>version: Fingerprint template version; currently available versions are 10.0 and 12.0;</p> <p>template: Fingerprint template content;</p> <p>templateNo: Fingerprint template number;</p> <p>templateNoIndex: Index corresponding to the fingerprint template.</p>
--------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.1.4.3 Delete BioTemplate by Personnel ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/bioTemplate/delete/{pin}?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pin: personnel ID, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.1.4.4 Delete Personnel BioTemplate Based on Personnel ID and BioTemplate ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/person/delete/{pin}{templateNo}?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	none

<b>Request Parameter Description</b>	pin: personnel ID, required; templateNo: BioTemplate ID, required
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

#### 2.1.4.5 Retrieve BioTemplate Information by Personnel ID--V2

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/bioTemplate/getFgListByPin?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	none
<b>Request Parameter Description</b>	pin: personnel ID, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "40280bb470e118510170e13f57dc001a",     "pin": "3170156",     "validType": "1",     "bioType": 1,     "version": "10",     "template": "TWITUzlxAAAeKikECAUHCc7QAAAcK2kBAAA     AhNcowCrDALI P3.....(Biotemplate)",     "templateNo": "6",</pre>

	<pre> "templateNoIndex": 0 } } </pre>
<b>Response Result Description</b>	<p>data: Personnel fingerprint data information;</p> <p>id: Personnel fingerprint ID;</p> <p>pin: Personnel identification number;</p> <p>validType: Validity status;</p> <p>bioType: Biometric template type.</p>

#### 2.1.4.6 Delete BioTemplate Based on Personnel ID --V2

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/bioTemplate/deleteByPin?access_token={apitoken}&pin={pin}
<b>Request Mode</b>	POST
<b>Request Content</b>	none
<b>Request Parameter Description</b>	pin: personnel ID, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

#### 2.1.4.7 Delete Personnel BioTemplate Based on Personnel ID and BioTemplate ID --V2

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/person/delete?access_token={apitoken}&pin={pin}&templateNo={templateNo}
<b>Request Mode</b>	POST

<b>Request Content</b>	none
<b>Request Parameter Description</b>	pin: personnel ID, required; templateNo: BioTemplate ID, required
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

## 2.2 Access Control

### 2.2.1 Access Control Device Interface

#### 2.2.1.1 Get Device List [device/acclist]

<b>Post Request URL</b>	http://serverIP:serverPort/api/device/acclist?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: pageno, required, the value must be greater than 0; pageSize: the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c3bbde1016c3bc2329200ff",       "sn": "0564140100195",       "name": "192.168.218.104",       "type": "inBIO160",       "state": "1", </pre>

	<pre> "module": "acc" }, {   "id": "402856aa6c3bbde1016c3bc211e50059",   "sn": "0566141900209",   "name": "192.168.218.11",   "type": "inBIO460",   "state": "1",   "module": "acc" } ] } </pre>
<b>Response Result Description</b>	<p>Data: device array list;          Id: device id;          Sn: device sn;          Name: device name;          state: device state, 1 enabled, 0 disabled;          Module: module, access, attendance, elevator;</p>

### 2.2.1.2 Get Device Info by SN [device/getAcc/{sn}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/device/getAcc?sn={sn}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Sn:device sn, required;
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "id": "402856aa6c3bbde1016c3bc2329200ff", </pre>

	<pre> "sn": "5662012052868", "name": "192.168.214.15", "type": "inibo460", "state": "0", "module": "acc" } } </pre>
<b>Response Result Description</b>	<p>Data: device object;</p> <p>Id: device id;</p> <p>Sn: device sn</p> <p>Name: device name;</p> <p>Type: device model type;</p> <p>state: device state, 1 enabled, 0 disabled;</p> <p>Module: module, access, attendance, elevator;</p>

### 2.2.1.3 Get Device List (With Total Number) [api/v2/device/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/device/acclist?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>pageNo: pageno, required, the value must be greater than 0;</p> <p>pageSize: the number of each page, required, must be greater than 0</p>
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 1,     "total": 1,     "data": [{ </pre>

	<pre>                 "id": "8a888e238478c3a6018478e40ee40a64",                 "sn": "CJI4200960003",                 "name": "10.8.14.142",                 "type": "SpeedFace-V5L",                 "status": "1",                 "module": "acc"             }},             "offset": 0,             "lastPage": true         }     }         </pre>
<p><b>Response Result Description</b></p>	<p>data: response result                  Page: request page number                  Size: data number of current request                  Total: total number                  data: device array list;                  Id: device id;                  Sn: device sn;                  Name: device name;                  state: device state, 1 enabled, 0 disabled;                  Module: module, access, attendance, elevator;</p>

## 2.2.2 Door Interface

### 2.2.2.1 Get the State of All Doors [door/all DoorState]

<p><b>Post Request URL</b></p>	<p>http://serverIP:serverPort/api/door/allDoorState?timestamp={timestamp}&amp;access_token={apitoken}</p>
<p><b>Request Mode</b></p>	<p>GET</p>
<p><b>Request Content</b></p>	<p>None</p>
<p><b>Request Parameter Description</b></p>	<p>Timestamp: timestamp, millisecond, such as 1494467276539, required; Access frequency cannot be too high, it is recommended to</p>

	obtain at least 10 seconds or more;
<b>Response Result</b>	<pre> { "code": 0, "message": "success", "data": [ { "id": door id, string; such as: "402856aa6c3bbde1016c3bc210d90016" "name": "door name", "deviceId": device id, string; such as: "402856aa6c3bbde1016c3bc211e5005a" "connect": device connection state (0 offline, 1online, 2 disabled), "sensor": sensor state (unknown, no sensor, off, open), "alarm": alarm state (none, alarm, timeout, be opened accidentally, tampering, duress password opening, duress fingerprint opening, duress opening, normal open alarm, low battery voltage, replace battery, illegal operation, backup power, unknown) "relay": relay state (off, open, unknown) }, { "id": "402856aa6c3bbde1016c3bc211e50059", "name": "192.168.218.11-1", "deviceId": "402856aa6c3bbde1016c3bc211e5005a", "connect": "1", "sensor": "no sensor", "alarm": "None", "relay": "off" }, { "id": "402856aa6c3bbde1016c3bc211e5005a", "name": "192.168.218.11-2", "deviceId": "402856aa6c3bbde1016c3bc211e5005a", "connect": "1", "sensor": "no sensor", "alarm": "none", </pre>

	<pre> "relay": "off" }, { "id": "402856aa6bff4b1a016bff5374800c0e", "name": "192.168.218.11-3", "deviceId": "402856aa6bff4b1a016bff5374800c0e", "connect": "1", "sensor": "no sensor", "alarm": "None", "relay": "off" }, { "id": "402856aa6c083827016c08582f8f0012", "name": "192.168.218.11-4", "deviceId": "402856aa6c3bbde1016c3bc211e5005a", "connect": "1", "sensor": "No sensor", "alarm": "None", "relay": "off" } ] } </pre>
<p><b>Response Result Description</b></p>	<p>Data: door state array List;</p> <p>Id: door id;</p> <p>Name: door name;</p> <p>DeviceId: device id;</p> <p>Connection: connection state, 0 offline, 1 online, 2 disabled;</p> <p>Sensor: sensor description (unknown, no sensor, off, open)</p> <p>Alarm: alarm description (none, alarm, timeout, be opened accidentally, tampering, duress password opening, duress fingerprint opening, duress opening, normal open alarm, low battery voltage, replace battery, illegal operation, backup power, unknown)</p> <p>relay: Relay state (off, open, unknown)</p>

### 2.2.2.2 Get Door State by Door Id [door/doorStateById]

<b>Post Request URL</b>	http://serverIP:serverPort/api/door/doorStateById?doorId=1&timestamp={timestamp}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Timestamp: timestamp, millisecond, such as 1494467276539, required; Access frequency cannot be too high, it is recommended to obtain at least 10 seconds or more;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": door id, string; such as: 402856aa6c083827016c08582f8f0012     "name": "door name",     "deviceId": device id, string; such as: "402856aa6c3bbde1016c3bc211e5005a"     "connect": device connection state,     "sensor": sensor state description,     "alarm": alarm state description,     "relay": relay state   } }</pre>
<b>Response Result Description</b>	<p>data: door state object</p> <p>id: door id;</p> <p>name: door name;</p> <p>deviceId: device id;</p> <p>connect: connection state (0 offline, 1 online, 2 disabled);</p> <p>sensor: sensor description (unknown, no sensor, off, open);</p> <p>alarm: alarm state (none, alarm, timeout, be opened accidentally, tampering, duress password opening, duress fingerprint opening, duress opening, normal open alarm, low battery voltage, replace battery, illegal operation, backup power, unknown)</p> <p>relay: Relay state (off, open, unknown)</p>

### 2.2.2.3 Get Door State by the Sn of Device [door/doorStateBy{Sn}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/door/doorStateBySn?deviceSn={sn}&timestamp={timestamp}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Timestamp: timestamp, millisecond, such as 1494467276539, required; Access frequency cannot be too high, it is recommended to obtain at least 10 seconds or more; Sn: device sn
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": door id, string; such as:     "402856aa6c083827016c08582f8f0012",     "name": "door name",     "deviceId": device id,     "connect": device state description,     "sensor": sensor state description,     "alarm": alarm state description     "relay": relay state description   } }</pre>
<b>Response Result Description</b>	<p>data: door state object;</p> <p>id: door id;</p> <p>name: door name;</p> <p>deviceId: device id;</p> <p>connect: device connection state (0 offline, 1 online, 2 disabled);</p> <p>sensor: sensor state (unknown, no sensor, off, open);</p> <p>alarm: alarm state (none, alarm, timeout, be opened accidentally, tampering, duress password opening, duress fingerprint opening,</p>

	duress opening, normal open alarm, low battery voltage, replace battery, illegal operation, backup power, unknown) relay: relay state (off, open, unknown)
--	---------------------------------------------------------------------------------------------------------------------------------------------------------------

#### 2.2.2.4 Get Door Info by Door id [door/get]

<b>Post Request URL</b>	http://serverIP:serverPort/api/door/get?id={id}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	id: door id;
<b>Response Result</b>	{ "code": 0, "message": "success", "data": { "id": "402856aa6c083827016c08582f8f0012", "name": "192.168.214.15-1", "deviceId": "402856aa6c3bbde1016c3bc211e5005a", } }
<b>Response Result Description</b>	data: door info object; id: doorid; name: door name; deviceId: deviceid;

### 2.2.2.5 Get Door List [door/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/door/list?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	PageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c083827016c08582f8f0012",       "name": "192.168.214.15-1",       "deviceId": "402856aa6c3bbde1016c3bc211e5005a",     },     {       "id": "402856aa6c08e7e4016c090be94a000a",       "name": "192.168.214.15-2",       "deviceId": "402856aa6c3bbde1016c3bc211e5005a",     },     {       "id": "402856aa6c08e7e4016c092400580021",       "name": "192.168.214.15-3",       "deviceId": "402856aa6c3bbde1016c3bc211e5005a",     },     {       "id": "402856aa6c3baabb016c3bb37365000a",       "name": "192.168.214.15-4",       "deviceId": "402856aa6c3baabb016c3bb37365000a"     }   ] }</pre>

	<pre> } ] } </pre>
<b>Response Result Description</b>	data: door array list; id: doorid; name: door name; deviceId: device id;

### 2.2.2.6 Remote Close by Door Id [door/remoteCloseById]

<b>Post Request URL</b>	http://serverIP:serverPort/api/door/remotecloseById?doorId={doorId}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	doorId: door id, required, string; such as: "402856aa6c3baabb016c3bb37365000a"
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.2.2.7 Remote Close by Door Name [door/remoteCloseByName]

<b>Post Request URL</b>	http://serverIP:serverPort/api/door/remotecloseByName?doorName={doorName}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	doorName: door name, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>

<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>
------------------------------------	--------------------------------------------------------------------------------------------

### 2.2.2.8 Remote Open by Door Id [door/remoteOpenById]

<b>Post Request URL</b>	http://serverIP:serverPort/api/door/remoteOpenById?doorId={doorId}&interval={interval}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	doorId: door id, required, string; such as: "402856aa6c3baabb016c3bb37365000a", Interval: door opening time, seconds, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.2.2.9 Remote Open by Door Name[door/remoteOpenByName]

<b>Post Request URL</b>	http://serverIP:serverPort/api/door/remoteOpenByName?doorName={doorName}&interval={interval}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	doorName: door name, required; interval: door opening time, seconds, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.2.2.10 Get Door List (With Total Number) [api/v2/door/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/door/list?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	PageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 1,     "total": 1,     "data": [{       "id": "8a888e238478c3a6018478e40f230aae",       "name": "10.8.14.142-1",       "deviceId": "8a888e238478c3a6018478e40ee40a64"     }],     "offset": 0,     "lastPage": true   } }</pre>
<b>Response Result Description</b>	<p>data: result;</p> <p>Page: current request page;</p> <p>Size: current data number of current request page;</p> <p>Total: total number;</p> <p>data: door array list;</p> <p>id: door id;</p>

	name: door name; deviceId: device id;
--	------------------------------------------

## 2.2.3 Reader Interface

### 2.2.3.1 Get Access Control Reader Info List[reader/acclist]

<b>Post Request URL</b>	http://serverIP:serverPort/api/reader/acclist?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	PageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c3baabb016c3bb37365000a",       "doorId": "402856aa6bff4b1a016bff5374800c0e",       "name": "192.168.214.15-1-in",       "readerNo": 1,       "readerState": 0     },     {       "id": "402856aa6c3c5063016c3cb3a0360005",       "doorId": "402856aa6bff4b1a016bff5374800c0e",       "name": "192.168.214.15-1-out",       "readerNo": 2,       "readerState": 1     }   ] }</pre>

	] }
<b>Response Result Description</b>	Data: reader array List; Id: reader id; DoorId: door id; Name: reader name; ReaderNo: readerNo; ReaderState: reader state, 0: in, 1: out;

### 2.2.3.2 Get Access Control Reader Info by Reader Id [reader/getAcc]

<b>Post Request URL</b>	http://serverIP:serverPort/api/reader/getAcc?id={id}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	id: reader id, required;
<b>Response Result</b>	{ "code": 0, "message": "success", "data": { "id": "402856aa6c3c5063016c3cb3a0360005", "doorId": "402856aa6bff4b1a016bff5374800c0e", "name": "192.168.214.15-1-in", "readerNo": 1, "readerState": 0 } }
<b>Response Result Description</b>	data: reader info object; id: reader id; doorId: doorid;

	name: reader name; readerNo: reader no; readState: reader state, 0: in, 1:out;
--	--------------------------------------------------------------------------------------

### 2.2.3.3 Get Access Control Reader Info List[api/v2/reader/list] (With Total Number)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/reader/list?pageNo=1&pageSize=2&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 2,     "total": 116,     "data": [       {         "id": "402856aa6c3baabb016c3bb37365000a",         "doorId": "402856aa6bff4b1a016bff5374800c0e",         "name": "192.168.214.15-1-in",         "readerNo": 1,         "readerState": 0       },       {         "id": "402856aa6c3c5063016c3cb3a0360005",         "doorId": "402856aa6bff4b1a016bff5374800c0e",         "name": "192.168.214.15-1-out",       }     ]   } } </pre>

	<pre>"readerNo": 2, "readerState": 1 } ] }</pre>
<b>Response Result Description</b>	<p>Data: reader array List;</p> <p>page: the page number of the current request;</p> <p>size: the number of data requested by the current page number;</p> <p>total: total number;</p> <p>Id: reader id;</p> <p>DoorId: door id;</p> <p>Name: reader name;</p> <p>ReaderNo: readerNo;</p> <p>ReaderState: reader state, 0: in, 1: out;</p>

## 2.2.4 Access Control Level Interface

### 2.2.4.1 Delete Person Level [accLevel/deleteLevel]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accLevel/deleteLevel?pin={pin}&levelIds={levelIds}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>Pin: pin, required</p> <p>Level Ids: level group id List, multiple comma separated, required, string; such as 402856aa6c3bbde1016c3bc2329200ff,402856aa6c3bbde1016c3bc232930100</p>
<b>Response Result</b>	Refer to public Response Results
<b>Response Result Description</b>	Public response parameter description

### 2.2.4.2 Get Level by Id [accLevel/get byId{id}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accLevel/getById/{id}?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Id: level group id, required, string; such as "402856aa6c3bbde1016c3bc2329200ff"
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "402856aa6c3bbde1016c3bc2329200ff",     "name": "public level group"   } }</pre>
<b>Response Result Description</b>	Data: level group object; Id: level group id; Name: level group name;

### 2.2.4.3 Get Level by Id [accLevel/get byId{id}]--V2

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/accLevel/getById?id={id}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Id: level group id, required, string; such as "402856aa6c3bbde1016c3bc2329200ff"
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",</pre>

	<pre>"data": {   "id": "402856aa6c3bbde1016c3bc2329200ff",   "name": "public level group" } }</pre>
<b>Response Result Description</b>	<p>Data: level group object;          Id: level group id;          Name: level group name;</p>

#### 2.2.4.4 Get Level by Name [accLevel/get ByName{name}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accLevel/getByName/{name}?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Name: level group name, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "402856aa6c3bbde1016c3bc2329200ff",     "name": "public level group"   } }</pre>
<b>Response Result Description</b>	<p>Data: level group object;          Id: level group id, string; such as '402856aa6c3bbde1016c3bc2329200ff'          Name: level group name;</p>

### 2.2.4.5 Get Level by Name [accLevel/get ByName{name}]-V2

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/accLevel/getByName?name={name}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Name: level group name, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "402856aa6c3bbde1016c3bc2329200ff",     "name": "public level group"   } }</pre>
<b>Response Result Description</b>	Data: level group object; Id: level group id, string; such as '402856aa6c3bbde1016c3bc2329200ff' Name: level group name;

### 2.2.4.6 Query the bound permission group according to the department [accLevel/getAccLevelListByDeptCode]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accLevel/getAccLevelListByDeptCode?deptCode={deptCode}&pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	deptCode: Department number, required; pageNo: Page number, required, must be greater than 0;

	pageSize: Number of items per page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "402856aa6c3bbde1016c3bc2329200ff",     "name": "General Permission Group"   } }</pre>
<b>Response Result Description</b>	<p>Data: Permission group object;</p> <p>Id: Permission group ID; such as: "402856aa6c3bbde1016c3bc2329200ff"</p> <p>Name: Permission Group Name;</p>

#### 2.2.4.7 Get Access Level/list [accLevel/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accLevel/list?pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c3bbde1016c3bc2329200ff",       "name": "public level group"     }   ] }</pre>

	] }
<b>Response Result Description</b>	Data: level group array list; Id: level group id, string; such as "402856aa6c3bbde1016c3bc2329200ff" Name: level group name;

#### 2.2.4.8 Sync Level Info to Device [accLevel/syncLevel]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accLevel/syncLevel?levelId={levelId}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Level Id: level group id, required, string; such as "402856aa6c3bbde1016c3bc2329200ff"
<b>Response Result</b>	Refer to public response result, for any other result please refer to Appendix- <a href="#">Error code</a>
<b>Response Result Description</b>	Public response parameter description

#### 2.2.4.9 Sync Person Level [accLevel/syncPerson]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accLevel/syncPerson?pin={pin}&levelIds={levelIds}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Level Ids: level group id list, multiple comma separated, required, string; such as "402856aa6c3bbde1016c3bc2329200ff" Pin: pin, required
<b>Response Result</b>	Refer to public response result, other can refer to Appendix-Error code

<b>Response Result Description</b>	Public response parameter description
------------------------------------	---------------------------------------

### 2.2.4.10 Get Access Level/List (With total number)[/api/v2/accLevel/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/accLevel/list?pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	PageNo: page no, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 1,     "total": 1,     "data": [{       "id": "8a888e238474d72d018474d8cba8065a",       "name": "General"     }],     "offset": 0,     "lastPage": true   } }</pre>
<b>Response Result Description</b>	<p>page: the page number of the current request;</p> <p>size: the amount of data requested for the current page number;</p> <p>total: the total number;</p> <p>data: access levels arrays;</p> <p>id: access level id; such as: "8a888e238474d72d018474d8cba8065a"</p> <p>name: access level name;</p>

### 2.2.4.11 Bulk Add/Edit Access Level Group[accLevel/addLevel]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accLevel/addLevel?access_token={api token}
<b>Request Mode</b>	POST
<b>Request Content</b>	[ { "areaName": "Area1", "name": "Access Level 1", "timeSegName": "TimeTable 1" }, { "areaName": "Area 2", "name": "Access Level 2", "timeSegName": "TimeTable 1" } ]
<b>Request Parameter Description</b>	name: Access Level Name, required; timeSegName: The selected time zone name, required; areaName: The selected area name, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

## 2.2.4.12 Bulk Add/Edit Access Control Groups and Add Doors

[accLevel/addLevelDoor]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accLevel/addLevelDoor?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	[ { "doorName": "10.8.14.162-1", "levelName": "General Access Level" }, { "doorName": "10.8.14.162-2", "levelName": "General Access Level" } ]
<b>Request Parameter Description</b>	levelName: Access Level Name, required; doorName: The name of the door that was added, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

## 2.2.4.13 Bulk Assign Access Level for Personnel [accLevel/addLevelPerson]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accLevel/addLevelPerson?levelIds={levelIds}&pin={pin}&access_token={apitoken}
<b>Request Mode</b>	POST

<b>Request Content</b>	None
<b>Request Parameter Description</b>	pin: Personnel ID, required;  levelIds: Access group ID set, multiple IDs separated by commas, required.
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

## 2.2.5 Access Control transaction Interface

### 2.2.5.1 Get Transaction by Device SN [transaction/device/{deviceSn}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/transaction/device/{deviceSn}?pageNo=1&pageSize=20&acc_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	PageNo: pageNo, required, the value must be greater than 0; PageSize: required, the value must be greater than 0, not more than 1000 per page; DeviceSn: device sn; required;
<b>Response Result</b>	{ "code": 0, "message": "success", "data": [ { "id": "402856aa6c3c5063016c3cb3a0360005", "eventTime": "2017-08-04 15:59:12", "pin": "", "name": "", "lastName": "",

	<pre> "deptName": "", "areaName": "area name", "cardNo": "", "devSn": "5662012052868", "verifyModeName": "others", "eventName": "enable device", "eventPointName": "", "readerName": "Main door-In Reader", "devName": "192.168.214.15", "logId": 11734 }, { "deptName": "", "areaName": "area name", "cardNo": "", "devSn": "5662012052868", "verifyModeName": "others", "eventName": "connect server successfully", "eventPointName": "", "readerName": "others", "devName": "192.168.214.15", "logId": 11733 } ] } </pre>
<p><b>Response Result Description</b></p>	<p>data: transaction array list;  id: "402856aa6c032ca3016c03351d2600fd";  eventTime: record device trigger time;</p>

	<p>pin: pin;</p> <p>name: person name;</p> <p>lastName: person name, (English);</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p>
--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.2.5.2 Get Transaction by Device SN (With Total Number)

[v2/transaction/device/{deviceSn}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/transaction/device/{deviceSn}?pageNo=1&pageSize=20&acc_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: pageNo, required, the value must be greater than 0;</p> <p>PageSize: required, the value must be greater than 0, not more than 1000 per page;</p> <p>DeviceSn: device sn;</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {</pre>

```
"page": 0,  
"size": 2,  
"total": 39,  
"data": [{  
    "id": "8a888e2385a383f80185a4ccbe870037",  
    "eventTime": "2023-01-12 15:05:56",  
    "pin": "57",  
    "name": "",  
    "lastName": "",  
    "deptName": "Department Name",  
    "areaName": "Area Name",  
    "cardNo": "",  
    "devSn": "7633223440032",  
    "verifyModeName": "Face",  
    "eventName": "Normal Verify Open",  
    "eventPointName": "10.8.14.210-1",  
    "readerName": "10.8.14.210-1-Out",  
    "accZone": null,  
    "devName": "10.8.14.210",  
    "logId": 161,  
    "eventNo": 0,  
    "eventLevel": 0,  
    "doorName": "10.8.14.210-1",  
    "capturePhotoBase64": ""  
},  
{  
    "id": "8a888e2385a383f80185a4ccbe840036",  
    "eventTime": "2023-01-12 15:05:54",  
    "pin": "57",  
    "name": "",  
    "lastName": "",  
    "deptName": "Department Name",  
    "areaName": "Area Name",  
    "cardNo": "",
```

	<pre> "devSn": "7633223440032", "verifyModeName": "Face", "eventName": "Normal Verify Open", "eventPointName": "10.8.14.210-1", "readerName": "10.8.14.210-1-Out", "accZone": null, "devName": "10.8.14.210", "logId": 160, "eventNo": 0, "eventLevel": 0, "doorName": "10.8.14.210-1", "capturePhotoBase64": ""     }   ],   "offset": 0,   "lastPage": false } } </pre>
<p><b>Response Result Description</b></p>	<p>data: result  page: the page number of the current request;  size: the number of data requested by the current page number;  total: total number;  data: transaction array list;  id: "402856aa6c032ca3016c03351d2600fd";  eventTime: record device trigger time;  pin: pin;  name: person name;  lastName: person name, (English);  deptName: department name;  areaName: area name;  cardNo: card no;  devSn: device sn;  verifyModeName: verification mode name;</p>

	<p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p> <p>eventNo: event number</p> <p>eventLevel: event level</p> <p>doorName: door name</p> <p>capturePhotoBase64: captured photo of the door (base64)</p>
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.2.5.3 Get Transaction by Device SN (With Total Number--V3)

[v2/transaction/device/{deviceSn}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v3/transaction/device?deviceSn={deviceSn}&pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: pageNo, required, the value must be greater than 0;</p> <p>PageSize: required, the value must be greater than 0, not more than 1000 per page;</p> <p>DeviceSn: device SN;</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 2, </pre>

```
"total": 39,  
  
"data": [{  
  
    "id": "8a888e2385a383f80185a4ccbe870037",  
    "eventTime": "2023-01-12 15:05:56",  
    "pin": "57",  
    "name": "",  
    "lastName": "",  
    "deptName": "Department Name",  
    "areaName": "Area Name",  
    "cardNo": "",  
    "devSn": "7633223440032",  
    "verifyModeName": "Face",  
    "eventName": "Normal Verify Open",  
    "eventPointName": "10.8.14.210-1",  
    "readerName": "10.8.14.210-1-Out",  
    "accZone": null,  
    "devName": "10.8.14.210",  
    "logId": 161,  
    "eventNo": 0,  
    "eventLevel": 0,  
    "doorName": "10.8.14.210-1",  
    "capturePhotoBase64": "",  
  
    },
```

```
{  
  
  "id": "8a888e2385a383f80185a4ccbe840036",  
  "eventTime": "2023-01-12 15:05:54",  
  "pin": "57",  
  "name": "",  
  "lastName": "",  
  "deptName": "Department Name",  
  "areaName": "Area Name",  
  "cardNo": "",  
  "devSn": "7633223440032",  
  "verifyModeName": "Face",  
  "eventName": "Normal Verify Open",  
  "eventPointName": "10.8.14.210-1",  
  "readerName": "10.8.14.210-1-Out",  
    "readerState": 1,  
  "accZone": null,  
  "devName": "10.8.14.210",  
  "logId": 160,  
  "eventNo": 0,  
  "eventLevel": 0,  
  "doorName": "10.8.14.210-1",  
  "capturePhotoBase64": "",  
    "doorNumber": 1
```

	<pre>         }       ],       "offset": 0,       "lastPage": false     }   } </pre>
<p><b>Response Result Description</b></p>	<p>data: result  page: the page number of the current request;  size: the number of data requested by the current page number;  total: total number;  data: transaction array list;  id: "402856aa6c032ca3016c03351d2600fd";  eventTime: record device trigger time;  pin: pin;  name: First name;  lastName: Last name;  deptName: department name;  areaName: area name;  cardNo: card number;  devSn: device SN;  verifyModeName: verification mode name;  eventName: transaction description;  eventPointName: transaction point name;  readerName: reader name;  devName: device name;  logId: transaction index value, there will be a value if device support it;  eventNo: event number  eventLevel: event level  doorName: door name  capturePhotoBase64: captured photo of the door (base64)</p>

## 2.2.5.4 Get Access Transaction List[transaction/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/transaction/list?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>personPin: personPin, [not required]</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c032ca3016c03362ec10104",       "eventTime": "2017-08-04 15:59:12",       "pin": "",       "name": "",       "lastName": "",       "deptName": "",       "areaName": " area name",       "cardNo": "",       "devSn": "5662012052868",       "verifyModeName": "others",       "eventName": "enable device",       "eventPointName": "",       "readerName": "others",       "accZone": null,       "devName": "192.168.214.15",       "logId": 11734     }   ] }</pre>

	<pre> }, {   "id": "402856aa6c082c51016c082ef4e70001",   "eventTime": "2017-08-04 15:58:49",   "pin": "",   "name": "",   "lastName": "",   "deptName": "",   "areaName": "area name",   "cardNo": "",   "devSn": "5662012052868",   "verifyModeName": "others",   "eventName": "connect server successfully",   "eventPointName": "",   "readerName": "others",   "accZone": null,   "devName": "192.168.214.15",   "logId": 11733 } ] } </pre>
<p><b>Response Result Description</b></p>	<p>data: transaction array list;</p> <p>id:5;</p> <p>eventTime: record device trigger time;</p> <p>pin: pin;</p> <p>name: person name;</p> <p>lastName: person name, (English);</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p>

	<p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>accZone: access area name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p>
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.2.5.5 Get Access Transaction List (With Total Number)[api/v2/transaction/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/transaction/list?personPin={personPin}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>personPin: personPin, [not required]</p> <p>startDate: start Date, [not required]</p> <p>endDate: end Date, [not required]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 2,     "total": 39,     "data": [{       "id": "8a888e2385a383f80185a4ccbe870037",       "eventTime": "2023-01-12 15:05:56",       "pin": "57",       "name": "",       "lastName": ""     }   ] }</pre>

```

"deptName": "Department Name",
"areaName": "Area Name",
"cardNo": "",
"devSn": "7633223440032",
"verifyModeName": "Face",
"eventName": "Normal Verify Open",
"eventPointName": "10.8.14.210-1",
"readerName": "10.8.14.210-1-Out",
"accZone": null,
"devName": "10.8.14.210",
"logId": 161,
"eventNo": 0,
"eventLevel": 0,
"doorName": "10.8.14.210-1",
"capturePhotoBase64": ""
},
{
"id": "8a888e2385a383f80185a4ccbe840036",
"eventTime": "2023-01-12 15:05:54",
"pin": "57",
"name": "",
"lastName": "",
"deptName": "Department Name",
"areaName": "Area Name",
"cardNo": "",
"devSn": "7633223440032",
"verifyModeName": "Face",
"eventName": "Normal Verify Open",
"eventPointName": "10.8.14.210-1",
"readerName": "10.8.14.210-1-Out",
"accZone": null,
"devName": "10.8.14.210",
"logId": 160,
"eventNo": 0,

```

	<pre>                 "eventLevel": 0,                 "doorName": "10.8.14.210-1",                 "capturePhotoBase64": ""             }         ],         "offset": 0,         "lastPage": false     } } </pre>
<p><b>Response Result Description</b></p>	<p>data: result</p> <p>page: the page number of the current request;</p> <p>size: the number of data requested by the current page number;</p> <p>total: total number;</p> <p>data: transaction array list;</p> <p>id: "402856aa6c032ca3016c03351d2600fd";</p> <p>eventTime: record device trigger time;</p> <p>pin: pin;</p> <p>name: person name;</p> <p>lastName: person name, (English);</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p> <p>eventNo: event number</p> <p>eventLevel: event level</p> <p>doorName: door name</p> <p>capturePhotoBase64: captured photo of the door (base64)</p>

## 2.2.5.6 Get Access Real-Time Monitoring Transaction[transaction/monitor]

<b>Post Request URL</b>	http://serverIP:serverPort/api/transaction/monitor?timestamp={timestamp}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Timestamp: timestamp millisecond, such as 1494467276539, required; Access frequency cannot be too high, it is recommended to obtain at least 10 seconds or more; there is no need to change after timestamp value set the first request, it is allowed to keep using the first request
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "eventTime": "2017-05-15 16:35:41",       "pin": "",       "areaName": " area name",       "cardNo": "9606650",       "devSn": "0564140100195",       "verifyModeName": "card or fingerprint",       "eventName": "card is not registered",       "eventPointName": "192.168.218.11-1",       "readerName": "192.168.218.11-1- out",       "devName": "192.168.218.11(0566141900209)"     }   ] }</pre>
<b>Response Result Description</b>	data:transaction monitoring record array List; eventTime: record device trigger time; pin:pin;

	areaName: area name; cardNo: card no; devSn: device sn; verifyModeName: verification mode name; eventName: transaction description; eventPointName: transaction point name; readerName: reader name; devName: device name;
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.2.5.7 Get Access Transaction by Pin [transaction/person/{pin}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/transaction/person/{pin}?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: page no, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page; pin: pin, required; startDate: Start Date, [not required] endDate: End Date, [not required]
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c17518e016c175385fd0001",       "eventTime": "2017-08-04 15:59:12",       "pin": "",       "name": "",       "lastName": "",       "deptName": ""     }   ] }</pre>

	<pre> "areaName": "area name", "cardNo": "", "devSn": "5662012052868", "verifyModeName": "others", "eventName": " enable device", "eventPointName": "", "readerName": "others", "devName": "192.168.214.15", "logId": 11734 }, { "id": 5, "eventTime": "2017-08-04 15:58:49", "pin": "", "name": "", "lastName": "", "deptName": "", "areaName": "area name", "cardNo": "", "devSn": "5662012052868", "verifyModeName": "others", "eventName": "connect server successfully", "eventPointName": "", "readerName": "others", "devName": "192.168.214.15", "logId": 11733 } ] } </pre>
<p><b>Response Result Description</b></p>	<p>data:transaction array list;  id:5;  eventTime: record device trigger time;  pin: pin;</p>

	<p>name: name;</p> <p>lastName: person name (English);</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>accZone: access area name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.2.5.8 Get Access Transaction by Pin (With Total Number)

[api/v2/transaction/person/{pin}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/transaction/person/{pin}?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>personPin: personPin, [not required]</p> <p>startDate: Start Date, [not required]</p> <p>endDate: End Date, [not required]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {</pre>

```

"page": 0,
"size": 2,
"total": 39,
"data": [{
    "id": "8a888e2385a383f80185a4ccbe870037",
    "eventTime": "2023-01-12 15:05:56",
    "pin": "57",
    "name": "",
    "lastName": "",
    "deptName": "Department Name",
    "areaName": "Area Name",
    "cardNo": "",
    "devSn": "7633223440032",
    "verifyModeName": "Face",
    "eventName": "Normal Verify Open",
    "eventPointName": "10.8.14.210-1",
    "readerName": "10.8.14.210-1-Out",
    "accZone": null,
    "devName": "10.8.14.210",
    "logId": 161,
    "eventNo": 0,
    "eventLevel": 0,
    "doorName": "10.8.14.210-1",
    "capturePhotoBase64": ""
  },
  {
    "id": "8a888e2385a383f80185a4ccbe840036",
    "eventTime": "2023-01-12 15:05:54",
    "pin": "57",
    "name": "",
    "lastName": "",
    "deptName": "Department Name",
    "areaName": "Area Name",
    "cardNo": "",

```

	<pre> "devSn": "7633223440032", "verifyModeName": "Face", "eventName": "Normal Verify Open", "eventPointName": "10.8.14.210-1", "readerName": "10.8.14.210-1-Out", "accZone": null, "devName": "10.8.14.210", "logId": 160, "eventNo": 0, "eventLevel": 0, "doorName": "10.8.14.210-1", "capturePhotoBase64": ""     }   ],   "offset": 0,   "lastPage": false } } </pre>
<p><b>Response Result Description</b></p>	<p>data: result</p> <p>page: the page number of the current request;</p> <p>size: the number of data requested by the current page number;</p> <p>total: total number;</p> <p>data: transaction array list;</p> <p>id: "402856aa6c032ca3016c03351d2600fd";</p> <p>eventTime: record device trigger time;</p> <p>pin: pin;</p> <p>name: person name;</p> <p>lastName: person name, (English);</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p>

	<p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p> <p>eventNo: event number</p> <p>eventLevel: event level</p> <p>doorName: door name</p> <p>capturePhotoBase64: captured photo of the door (base64)</p>
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.2.5.9 Get Access Transaction List by pin--V3 (With Total Number)

[api/v2/transaction/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v3/transaction/person?pageNo=1&pageSize=20&pin={pin}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>Pin:Person ID, [not required]</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p>
<b>Response Result</b>	<pre>{   "code":0,   "message":"success",   "data":{     "page":0,     "size":1, </pre>

```
"total":2,
"data":[
  {
    "id":"8a888e2385adc3af0185af769dab0031",
    "eventTime":"2023-01-14 16:47:36",
    "pin":"59",
    "name": "",
    "lastName": "",
    "deptName":"Department Name",
    "areaName":"Area Name",
    "cardNo": "",
    "devSn":"7633223440032",
    "verifyModeName":"Palm",
    "eventName":"Nomal verify open the door",
    "eventPointName":"10.8.14.210-1",
    "readerName":"10.8.14.210-1-Out",
    "accZone":null,
    "devName":"10.8.14.210",
    "logId":167,
    "eventNo":0,
    "eventLevel":0,
    "doorName":"10.8.14.210-1",
    "capturePhotoBase64": ""
```

	<pre>         }       ],       "offset":0,       "lastPage":false     }   } </pre>
<p><b>Response Result Description</b></p>	<p>data: result</p> <p>page: the page number of the current request;</p> <p>size: the number of data requested by the current page number;</p> <p>total: total number;</p> <p>data: transaction array list;</p> <p>id: "402856aa6c032ca3016c03351d2600fd";</p> <p>eventTime: record device trigger time;</p> <p>pin: personnel ID;</p> <p>name: First name;</p> <p>lastName: Last name</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card number;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>accZone: Access zone name</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p> <p>eventNo: Event ID</p> <p>eventLevel: Event Level</p>

	doorName: Door Name
	capturePhotoBase64: Captured Photo (base64)

### 2.2.5.10 Get Personnel First In and Last Out Record by Pin

[transaction/firstInAndLastOut/{pin}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/transaction/firstInAndLastOut/{pin}?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p> <p>pin: pin, required;</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "40285ab374521d14017452820eb50005",       "pin": "147",       "name": "",       "lastName": "",       "readerNameIn":       "192.168.218.183-1-Out",       "firstInTime": "2020-09-03 13:48:07",       "readerNameOut": null,       "lastOutTime": null,       "deptCode": "1",     }   ] }</pre>

	<pre> "deptName": "Department Name"     }   ] } </pre>
<b>Response Result Description</b>	<p>data:transaction array list;</p> <p>id:5;</p> <p>pin: pin;</p> <p>name: name;</p> <p>lastName: person name (English);</p> <p>deptCode: department code;</p> <p>deptName: department name;</p> <p>readerNameIn: First In reader name;</p> <p>firstInTime: First In Time;</p> <p>readerNameOut: Last Out reader name;</p> <p>lastOutTime: Last Out Time;</p>

### 2.2.5.11 Get Personnel First In and Last Out Record by Pin (With Total Number)

[api/v2/transaction/firstInAndLastOut/{pin}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/transaction/firstInAndLastOut/{pin}?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p> <p>pin: pin, required;</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p>
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success", </pre>

	<pre> "data": {   "page": 0,   "size": 2,   "total": 1,   "data": [{     "id": "8a888e238599329401859ae866b4005d",     "pin": "50",     "name": "",     "lastName": "",     "readerNameIn": "10.8.14.142-1-Out",     "firstInTime": "2012-11-04 00:05:58",     "readerNameOut": "10.8.14.142-1-Out",     "lastOutTime": "2012-11-04 03:11:52",     "deptCode": "1",     "deptName": "Department Name"   }],   "offset": 0,   "lastPage": true } </pre>
<p><b>Response Result Description</b></p>	<p>data: result  page: the page number of the current request;  size: the number of data requested by the current page number;  total: total number;  data: transaction array list;  id:5;  pin: pin;  name: name;  lastName: person name (English);  deptCode: department code;  deptName: department name;  readerNameIn: First In reader name;  firstInTime: First In Time;</p>

	readerNameOut: Last Out reader name; lastOutTime: Last Out Time;
--	---------------------------------------------------------------------

### 2.2.5.12 Get Personnel First In and Last Out Record by Pin (With Total Number -V3) [api/v2/transaction/firstInAndLastOut/{pin}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/transaction/firstInAndLastOut?pageNo=1&pageSize=20&pin={pin}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p> <p>pin: pin, required;</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 2,     "total": 1,     "data": [{       "id": "8a888e238599329401859ae866b4005d",       "pin": "50",       "name": ""     }   ] }</pre>

	<pre> "lastName": "", "readerNameIn": "10.8.14.142-1-Out", "firstInTime": "2012-11-04 00:05:58", "readerNameOut": "10.8.14.142-1-Out", "lastOutTime": "2012-11-04 03:11:52", "deptCode": "1", "deptName": "Department Name" }}, "offset": 0, "lastPage": true } } </pre>
<p><b>Response Result Description</b></p>	<p>data: result  page: the page number of the current request;  size: the number of data requested by the current page number;  total: total number;  data:transaction array list;  id:"40285ab374521d14017452820eb50005";  pin: Personnel ID;  name: First name;  lastName: Last name;  deptCode: department ID;  deptName: department name;  readerNameIn: First In reader name;  firstInTime: First In Time;  readerNameOut: Last Out reader name;  lastOutTime: Last Out Time;</p>

### 2.2.5.13 Obtain Door Transaction, Daily Transaction, Personnel Access Transactions [transaction/getDoorTransactions]

<b>Post Request URL</b>	http://serverIP:serverPort/api/transaction/getDoorTransactions?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "devSn": "5662012052868",   "startTime": "2021-11-19 18:00:00",   "endTime": "2021-11-19 18:00:00",   "filter": "Name",   "type": "normal",   "eventNo": "0",   "personPin": "320022",   "pageNo": "1",   "pageSize": "100", }</pre>
<b>Request Parameter Description</b>	<p>devsn: Device Serial Number [Not required]</p> <p>startTime: Start Time [Not required]</p> <p>endTime: End Time [Not required]</p> <p>filter: Filter conditions [Not required, filter by name, last name, PIN] (Fuzzy query)</p> <p>type: Event Type [Not required, alarm events, normal event, warning events]</p> <p>eventNo: Events Number [Not required]</p> <p>personPin: personnel ID number[Not required] [Accurate Query]</p> <p>page: page number</p> <p>pageSize: record number of current page</p>
<b>Response Result</b>	<pre>{   "ret": "ok",   "msg": "operation succeed",   "data": [     {</pre>

```

    "id": "402856a7695c716301695c81290d0006",
    "eventNo": 8,
    "eventName": "remote opening",
    "eventTime": "2019-03-08 16:52:38",
    "doorName": "192.168.214.197-1",
    "personPin": "",
    "personName": "",
    "areaName": "Area Name",
    "personLastName": "",
    "eventLevel": "0"
  },
  {
    "id": "402856a7695c716301695c7fe7620004",
    "eventNo": 8,
    "eventName": "remote opening",
    "eventTime": "2019-03-08 16:51:14",
    "doorName": "192.168.214.197-1",
    "personPin": "",
    "personName": "",
    "areaName": "Area Name",
    "personLastName": "",
    "eventLevel": "0"
  }
],
"i18nArgs": null,
"page": 1,
"pageSize": 10,
"total": 2,
"success": true
}
{
  "ret": "fail",
  "msg": "failed info",
  "data": null,

```

	<pre>"i18nArgs": null }</pre>
<b>Response Result Description</b>	<p>data: transaction array list;</p> <p>Id: event id</p> <p>eventNo: event number</p> <p>eventName: event name</p> <p>eventTime: event time</p> <p>eventLevel: 0 normal 1 warning 2 alarm</p>

### 2.2.5.14 Obtain Door Transaction, Daily Transaction, Personnel Access Transactions (With Total Number) [api/v2/transaction/getDoorTransactions]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/transaction/getDoorTransactions?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>"content": {   "devSn": "7633223440032",   "startTime": "2023-1-13 18:00:00",   "endTime": "2023-1-14 18:00:00",   "filter": "",   "type": "normal",   "eventNo": "0",   "personPin": "59",   "pageNo": "1",   "pageSize": "2" }</pre>
<b>Request Parameter Description</b>	<p>devsn: Device Serial Number [Not required]</p> <p>startTime: Start Time [Not required]</p> <p>endTime: End Time [Not required]</p> <p>filter: Filter conditions [Not required, filter by name, last name, PIN] (Fuzzy query)</p> <p>type: Event Type [Not required, alarm events, normal event, warning events]</p>

	<p>eventNo: Events Number [Not required]          personPin: personnel ID number[Not required] [Accurate Query]          page: page number          pageSize: record number of current page</p>
<p><b>Response Result</b></p>	<pre> {   "ret": "ok",   "msg": "Operation Succeed",   "data": {     "page": 0,     "size": 2,     "total": 2,     "data": [{       "id": "8a888e2385adc3af0185af769dab0031",       "eventTime": "2023-01-14 16:47:36",       "pin": "59",       "name": "",       "lastName": "",       "deptName": "Department Name",       "areaName": "Area Name",       "cardNo": "",       "devSn": "7633223440032",       "verifyModeName": "Palm",       "eventName": "Normal Verify Open",       "eventPointName": "10.8.14.210-1",       "readerName": "10.8.14.210-1-Out",       "accZone": null,       "devName": "10.8.14.210",       "logId": 167,       "eventNo": 0,       "eventLevel": 0,       "doorName": "10.8.14.210-1",       "capturePhotoBase64": ""     }   ], }</pre>

```
{
  "id": "8a888e2385adc3af0185af769d9b0030",
  "eventTime": "2023-01-14 16:47:35",
  "pin": "59",
  "name": "",
  "lastName": "",
  "deptName": "Department Name",
  "areaName": "Area Name",
  "cardNo": "",
  "devSn": "7633223440032",
  "verifyModeName": "Face",
  "eventName": "Normal Verify Open",
  "eventPointName": "10.8.14.210-1",
  "readerName": "10.8.14.210-1-Out",
  "accZone": null,
  "devName": "10.8.14.210",
  "logId": 166,
  "eventNo": 0,
  "eventLevel": 0,
  "doorName": "10.8.14.210-1",
  "capturePhotoBase64": ""
}
],
"offset": 0,
"lastPage": true
},
"i18nArgs": null,
"success": true
}
{
  "ret": "fail",
  "msg": "failed message",
  "data": null,
  "i18nArgs": null
}
```

	} }
<b>Response Result Description</b>	<p>data: result</p> <p>page: the page number of the current request;</p> <p>size: the number of data requested by the current page number;</p> <p>total: total number;</p> <p>data: transaction array list;</p> <p>id: "402856aa6c032ca3016c03351d2600fd";</p> <p>eventTime: record device trigger time;</p> <p>pin: pin;</p> <p>name: person name;</p> <p>lastName: person name, (English);</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p> <p>eventNo: event number</p> <p>eventLevel: event level</p> <p>doorName: door name</p> <p>capturePhotoBase64: captured photo of the door (base64)</p>

### 2.2.5.15 Get Doors Transaction Details [transaction/getDoorTransactionDetail]

<b>Post Request URL</b>	http://serverIP:serverPort/api/transaction/getDoorTransactionDetail?access_token={apitoken}
<b>Request Mode</b>	POST

<b>Request Content</b>	<pre>{   "id": "402856a7695c716301695c81290d0006", }</pre>
<b>Request Parameter Description</b>	<pre>{   "content":{     id: event ID   } }</pre>
<b>Response Result</b>	<pre>{   "ret": "ok",   "msg": "operation succeed",   "data": {     "id": "402856a7694641ff0169468394b2000d",     "eventNo": 8, event id     "eventName": "remote opening", event name     "eventTime": "2019-03-04 10:23:40", event time     "doorName": null, door name     "personPin": null, personnel ID     "personName": null, First Name     "areaName": "Area 1", Area Name     "personLastName": null, Last Name     "devName": "192.168.214.197", Device Name     "cardNo": null, Card Number     "readerName": "others", reader name     "verifyModeName": "others" Verification mode   },   "i18nArgs": null,   "success": true } {   "ret": "fail",   "msg": "failed info",   "data": null,</pre>

	<pre>"i18nArgs": null }</pre>
<b>Response Result Description</b>	<p>data:transaction array list;</p> <p>id: event id;</p> <p>eventNo: event number</p> <p>eventName: event name</p> <p>eventTime: event time</p>

### 2.2.5.16 Get Access Transactions by Event Id [api/transaction/getById/{id}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/transaction/getById/{id}?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	id: event id, required; eg: "8a888e8f83f376e50183f4019fb70bcf"
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "8a888e8f83f376e50183f4019fb70bcf",     "eventTime": "2022-10-17 17:53:22",     "pin": "2210174",     "name": "Leo",     "lastName": "Hou",     "deptName": "RD",     "areaName": "Area",     "cardNo": "",     "devSn": "CJI2194860001",     "verifyModeName": "Face",     "eventName": "Normal Verify Open",     "eventPointName": "10.8.14.181-1",     "readerName": "10.8.14.181-1-out",   } }</pre>

	<pre> "accZone": null, "devName": "10.8.14.181", "logId": 4521, "eventNo": 0, "eventLevel": 0, "doorName": "10.8.14.181-1", "capturePhotoBase64": "/9j/4AAQSkZJRgAB..."     }     }         </pre>
<p><b>Response Result Description</b></p>	<p>id: "402856aa6c032ca3016c03351d2600fd";</p> <p>eventTime: record device trigger time;</p> <p>pin: pin;</p> <p>name: person name;</p> <p>lastName: person name, (English);</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p> <p>eventNo: event number</p> <p>eventLevel: event level</p> <p>doorName: door name</p> <p>capturePhotoBase64: captured photo of the door (base64)</p>

### 2.2.5.17 Get Access Transactions by Event Id (With Total Number)

[api/transaction/getById/{id}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/transaction/getById?id={id}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	id: event id, required; eg: "8a888e8f83f376e50183f4019fb70bcf"
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "8a888e8f83f376e50183f4019fb70bcf",     "eventTime": "2022-10-17 17:53:22",     "pin": "2210174",     "name": "Leo",     "lastName": "Hou",     "deptName": "RD",     "areaName": "Area",     "cardNo": "",     "devSn": "CJI2194860001",     "verifyModeName": "Face",     "eventName": "Normal Verify Open",     "eventPointName": "10.8.14.181-1",     "readerName": "10.8.14.181-1-out",     "readerState": 1,     "accZone": null,     "devName": "10.8.14.181",     "logId": 4521,   } }</pre>

	<pre> "eventNo": 0,  "eventLevel": 0,  "doorName": "10.8.14.181-1",  "doorNumber": 1  "capturePhotoBase64": "/9j/4AAQSkZJRgAB..."  }  } </pre>
<p><b>Response Result Description</b></p>	<p>id: "402856aa6c032ca3016c03351d2600fd";</p> <p>eventTime: record device trigger time;</p> <p>pin: personnel id;</p> <p>name: first name;</p> <p>lastName: last name</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>accZone: access area name</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p> <p>eventNo: event id</p> <p>eventLevel: event level,0:normal, 1:abnormal , 2:Alarm</p> <p>doorName: door name</p> <p>capturePhotoBase64: captured photo of the door (base64)</p>

## 2.2.6 Advanced Access Control Interface

### 2.2.6.1 Retrieve Personnel by Area Code[accAdvanced/getWholsInsideByZone]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accAdvanced/getWholsInsideByZone?code={code}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	code: Area ID, required
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "zoneId": "8a889006909539dd0190955d9a5a0fa8",       "zoneName": "11",       "pin": "20"     }   ] }</pre>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.2.6.2 Retrieve a Collection of Personnel by Area Code

#### [accAdvanced/getWholsInsideZoneList]

<b>Post Request URL</b>	http://serverIP:serverPort/api/accAdvanced/getWholsInsideByZoneList?pageNo={pageNo}&pageSize={pageSize}&access_token={apito
-------------------------	-----------------------------------------------------------------------------------------------------------------------------

	ken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	PageNo: page no, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 2,     "total": 5,     "data": [       {         "zoneId": "8a888fe39319b9ee01931e657ac60c1e",         "zoneName": "ZONE1",         "pin": "10"       },       {         "zoneId": "8a888fe39319b9ee01931e657ac60c1e",         "zoneName": "ZONE1",         "pin": "77889"       }     ]   },   "offset": 0,   "lastPage": false }</pre>

<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>
------------------------------------	--------------------------------------------------------------------------------------------

## 2.3 Time & Attendance

### 2.3.1 Person Interface of Attendance Area

#### 2.3.1.1 Delete Area Person [attAreaPerson/delete]

<b>Post Request URL</b>	http://serverIP:serverPort/api/attAreaPerson/delete?access_token={apitoken}
<b>Request Mode</b>	DELETE
<b>Request Content</b>	{ "code": "1", "pins": "5" }
<b>Request Parameter Description</b>	Code: area code, required; Pins: pin, required
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

#### 2.3.1.2 Set Area Person [attAreaPerson/set]

<b>Post Request URL</b>	http://serverIP:serverPort/api/attAreaPerson/set?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	{ "code": "1",

	<pre>"pins": ["5"] }</pre>
<b>Request Parameter Description</b>	Code: area code, required; Pins: pin, required
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.3.1.3 Get Area Object by Area Code[area/get/{code}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/attAreaPerson/area/get/{code}?code=1&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Code: department code, required;
<b>Response Result</b>	<pre>{ "code": 0, "message": "success", "data": { "id": 1, "code": "402856aa6c3bbde1016c3bc2329200ff ", "name": "area name" } }</pre>
<b>Response Result Description</b>	Data: area object; Id: area id; Code: area code; name: area name;

### 2.3.1.4 Get Area Object by Area Code[area/get/{code}] --V2

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/attAreaPerson/area/get?code=1&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Code: department code, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": 1,     "code": "402856aa6c3bbde1016c3bc2329200ff ",     "name": "area name"   } }</pre>
<b>Response Result Description</b>	Data: area object; Id: area id; Code: area code; name: area name;

### 2.3.1.5 Get Area List [area/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/attAreaPerson/area/list?pageNo=1&pageSize=5&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: page, required, must be greater than 0 pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	{

	<pre> "code": 0, "message": "success", "data": [ { "id": "402856aa6c3bbde1016c3bc2329200ff", "code": "1", "name": "area name" }, { "id": "402856aa6c3bbde1016c3bc211e5005a", "code": "2", "name": "2" } ] } </pre>
<b>Response Result Description</b>	Data: area object List; Id: area id; code: area code; name: area name;

### 2.3.1.6 Get Area List -Total Quantity[attAreaPerson/area/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/attAreaPerson/area/list?pageNo=1&pageSize=5&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: page, required, must be greater than 0 pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre> { "code": 0, </pre>

	<pre> "message": "success",  "data": {    "page": 0,    "size": 1,    "total": 1,    "data": [{      "id": "402856aa6c3bbde1016c3bc211e50059",      "code": "1",      "name": "Area Name"    }],    "offset": 0,    "lastPage": false  }  } </pre>
<b>Response Result Description</b>	Data: area object List; Id: area id; code: area code; name: area name;

## 2.3.2 Edit Attendance Personnel Information

### 2.3.2.1 Edit Personal Information

<b>Post Request URL</b>	http://serverIP:serverPort /api/attPerson/editAttPerson?access_token={apitoken}
<b>Request Mode</b>	POST

<p><b>Request Content</b></p>	<pre>{     "isAttendance": false,     "perDevAuth": 0,     "personPin": 1,     "verifyMode": 0 }</pre>
<p><b>Request Parameter Description</b></p>	<p>isAttendance (boolean): Attendance or not (true is Yes, false is Not)</p> <p>perDevAuth (integer): Device operation authority (0 is ordinary person, 2 is registrar, 6 is administrators, 14 is super administrators)</p> <p>personPin (string): personal number</p> <p>verifyMode (integer): Verification method (0=Automatic recognition, 1=Fingerprint only, 2=User ID only, 3=Password only, 4=Card only, 5=Fingerprint or password, 6=Fingerprint or card, 7=Card or password, 8=User ID and fingerprint, 9=Fingerprint and password, 10=Card and fingerprint, 11=Card and password, 12=Fingerprint and password and card, 13=User ID and fingerprint and password, 14=(User ID and fingerprint)or(card and fingerprint), 15=Face, 16=Face and fingerprint, 17=Face and password, 18=Face and card, 19=Face and fingerprint and password, 20=Face and fingerprint and password, 21=Finger vein, 22=Finger vein and password, 23=Finger vein and card, 24=Finger vein and password and card, 25=Palm prints, 26=Palm prints and card, 27=Palm prints and face, 28=Palm</p>

	prints and fingerprint , 29=Palm prints and fingerprint and face)
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

## 2.3.3 Att Device Interface

### 2.3.3.1 Get Att Device [device/getAtt]

<b>Post Request URL</b>	http://serverIP:serverPort/api/device/getAtt?sn={sn}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	sn: Device Serial Number,[required]
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "402856aa6c3bbde1016c3bc210d90016",     "sn": "CEXB193160012",     "name": "CEXB193160012",     "type": "ZK-S1007",     "status": "0",     "module": "att"   } }</pre>
<b>Response Result Description</b>	data: Device object; id: Device id; sn: Device Serial Number; name: Device Name;

	type: Device Model Name; statu: Device Status, 1 Enable, 0 Disabled; module: Module, acc AccessControl, att Attendance, ele Elevator;
--	---------------------------------------------------------------------------------------------------------------------------------------------

## 2.3.4 Apply for Abnormal Attendance

### 2.3.4.1 Apply for Re-clock-in

<b>Post Request URL</b>	http://serverIP:serverPort/api/attApply/applySign?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "personPin": 1,   "remark": "api",   "signDatetime": "2021-11-19 18:00:00" }</pre>
<b>Request Parameter Description</b>	personPin (string): personal number remark (string): remarks signDatetime(string): Re-clock-in time
<b>Response Result</b>	<pre>{   "code":0,   "message":"success",   "data":null }</pre>
<b>Response Result</b>	

<b>Description</b>	
--------------------	--

### 2.3.4.2 Apply for Overtime

<b>Post Request URL</b>	http://serverIP:serverPort/api/attApply/applyOvertime?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "endDatetime": "2021-11-19 18:00:00",   "personPin": 1,   "remark": "api",   "startDatetime": "2021-11-19 13:00:00" }</pre>
<b>Request Parameter Description</b>	<p>endDatetime (string): End Time</p> <p>personPin (string): Personal Number</p> <p>remark (string): Remarks</p> <p>startDatetime (string): start time</p>
<b>Response Result</b>	<pre>{   "code":0,   "message":"success",   "data":null }</pre>
<b>Response Result Description</b>	

### 2.3.4.3 Ask for Leave

<b>Post Request URL</b>	http://serverIP:serverPort/api/attApply/applyLeave?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "endDatetime": "2021-11-19 18:00:00",   "leaveImagePath": "string",   "leaveTypeNo": "L1",   "personPin": 1,   "remark": "api",   "startDatetime": "2021-11-19 13:00:00" }</pre>
<b>Request Parameter Description</b>	<p>endDatetime (string): End time</p> <p>leaveImagePath (string): Leave photo path (For details/api/attApply/uploadLeaveImage)</p> <p>leaveTypeNo (string): Leave type number(For details/api/attApply/getLeaveTypeList)</p> <p>personPin (string): Personal number</p> <p>remark (string): Remark</p> <p>startDatetime (string): Start time</p>
<b>Response Result</b>	<pre>{   "code": 0,</pre>

	<pre>"message": "success", "data": null }</pre>
<b>Response Result Description</b>	

#### 2.3.4.4 Get Leave Type List

<b>Post Request URL</b>	http://serverIP:serverPort/api/attApply/getLeaveTypeList?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	
<b>Request Parameter Description</b>	
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [{     "leaveTypeNo": "L1",     "leaveTypeName": "personal leave"   },   {     "leaveTypeNo": "L2",     "leaveTypeName": "marriage leave"   }   ] }</pre>

	}
<b>Response Result Description</b>	leaveTypeNo: Leave type number leaveTypeName: Leave type name

### 2.3.4.5 Upload Leave Picture

<b>Post Request URL</b>	http://serverIP:serverPort/api/attApply/uploadLeavelImage?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	{ "base64Image": "base64Image" }
<b>Request Parameter Description</b>	base64Image (string): picture Base64code
<b>Response Result</b>	{ "code":0, "message":"success", "data":"/fileresource/att/leave/image/2021-12-03/1638503394663.jpg" }
<b>Response Result Description</b>	data: picture path

### 2.3.4.6 Calculate the Total Duration of Leave

<b>Post Request URL</b>	http://serverIP:serverPort/api/attApply/getApplyTimeLong?access_token={apitoken}
-------------------------	----------------------------------------------------------------------------------

<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "endDatetime": "2021-11-19 18:00:00",   "leaveImagePath": "string",   "leaveTypeNo": "L1",   "personPin": 1,   "remark": "api",   "startDatetime": "2021-11-19 13:00:00" }</pre>
<b>Request Parameter Description</b>	<p>endDatetime (string): End time</p> <p>leaveImagePath (string, optional): Leave photo path (For details/api/attApply/uploadLeaveImage)</p> <p>leaveTypeNo (string): Leave type number (For details/api/attApply/getLeaveTypeList)</p> <p>personPin (string): personal number</p> <p>remark (string,): Remark</p> <p>startDatetime (string): Start time</p>
<b>Response Result</b>	{ "code": 0, "message": "success", "data": null }
<b>Response Result Description</b>	

## 2.3.5 Att Transaction Interface

### 2.3.5.1 Get Att Transaction [transaction/listAttTransaction]

<b>Post Request URL</b>	http://serverIP:serverPort/api/transaction/listAttTransaction?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>personPin: personPin, [not required]</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, the value must be greater than 0, not more than 1000 per page;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c3c5063016c3cb5a88f001b",       "eventTime": "2017-08-04 15:59:12",       "pin": "",       "name": "",       "lastName": "",       "deptName": "",       "areaName": "areaname",       "cardNo": "",       "devSn": "5662012052868",       "verifyModeName": "others",       "eventName": "enable device",       "eventPointName": "",       "readerName": "others",     }   ] }</pre>

	<pre> "accZone": null, "devName": "192.168.214.15", "logId": 11734 }, { "id": "402856aa6c3c5063016c3cb5f7150021", "eventTime": "2017-08-04 15:58:49", "pin": "", "name": "", "lastName": "", "deptName": "", "areaName": "areaname", "cardNo": "", "devSn": "5662012052868", "verifyModeName": "others", "eventName": " connected to the server successfully", "eventPointName": "", "readerName": "others", "accZone": null, "devName": "192.168.214.15", "logId": 11733 } ] } </pre>
<p><b>Response Result Description</b></p>	<p>data: transaction array List;  id: "402856aa6c3c5063016c3cb5cfa2001d";  eventTime: Record device trigger time;  pin: pin;  name: name;  lastName: person name (English);  deptName: department name;  areaName: area name;  cardNo: card no;</p>

	<p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>accZone: access area name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.3.5.2 Get Att Transaction [transaction/listAttTransaction](V2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/transaction/listAttTransaction?personPin={personPin}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>personPin: personPin, [not required,Used for querying by personnel number]</p> <p>startDate: startDate, [not required,Used for querying based on time, together with the end time]</p> <p>endDate: endDate, [not required,Used for querying by time, together with the start time]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, the value must be greater than 0, not more than 1000 per page;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 2,     "total": 2,     "data": [ {</pre>

	<pre> "id": " 402856aa6c3c5063016c3cb5f7150021 " , "eventTime": "2017-08-04 15:58:49", "pin": "", "name": "", "lastName": "", "deptName": "", "areaName": "area name", "cardNo": "", "devSn": "5662012052868", "verifyModeName": "Other", "eventName": "Successfully connected to the server", "eventPointName": "Attendance point name", "readerName": "Other", "accZone": null, "devName": "192.168.214.15", "logId": 11733 }, ], "offset": 0, "lastPage": true }, } </pre>
<p><b>Response Result Description</b></p>	<p>data: transaction array List;  id: "402856aa6c3c5063016c3cb5cfa2001d";  eventTime: Record device trigger time;  pin: pin;  name: name;  lastName: person name (English);  deptName: department name;  areaName: area name;  cardNo: card no;  devSn: device sn;  verifyModeName: verification mode name;</p>

	<p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>accZone: access area name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 2.4 Offline Consumption

### 2.4.1 Offline Consumption and Consumption Record Interface

#### 2.4.1.1 Obtain the Offline Consumption Records

<b>Post Request URL</b>	http://serverIP:serverPort/api/transaction/listPosTransaction?personPin={personPin}&deptCode={deptCode}&startDate={ startDate}&endDate={ endDate}&pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	not have
<b>Request Parameter Description</b>	<p>personPin: Personnel number (not required);</p> <p>deptCode: Department number (not required);</p> <p>startDate: Start time of the query record (not required);</p> <p>endDate: End time of query record (not required);</p> <p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize The number of articles on each page, required, must be greater than 0;</p>
<b>Response Results</b>	{ "code": 0,

```
"message": "success",

"data": {

"page": 0,

"size": 20,

"total": 1,

"data": [

{

"id": "4028b3b2872b638601872b6e6b630012",

"posTime": "2023-4-7 10:20:41",

"personPin": "1234568",

"personName": "123",

"personLastName": "",

"sysCardNo": "1234568",

"cardStatus": "valid",

"cardSerialNum": 3,

"deptCode": 1,

"deptName": "Department Name",

"card": 356373893,

"typeName": 6,

"money": 12.34,

"subMoney": 1.56,

"balance": 100.01,

"subBalance": "9.98",
```

	<pre> " posMode": 1,  "hallingCode": "1",  "hallingName": "Headquarters",  "mealCode": "1",  "mealName": "Breakfast",  "devSn": "10001",  "devAlias": "canteen consumer machine",      " devSerialNum": 1088  }  ],  "offset": 0,  "lastPage": true  }  } </pre>
<p><b>Response Result Description</b></p>	<p>posTime Consumption time</p> <p>personPin Personnel number</p> <p>personName Name of the person</p> <p>personLastName Persons last name</p> <p>sysCardNo Card account number</p> <p>cardStatus Card status (1 valid 3 loss reporting 4 consumption expired 5 shutdown)</p> <p>cardSerialNum Card serial number</p> <p>deptCode Department number</p> <p>deptName Department name</p>

Card, the physical card number

typeName Type name (6. Consumption 8. replenishment 9. Error correction 10. Count 11. bookkeeping)

money Consumption amount (yuan, 2 decimal places)

subMoney Subsidy wallet consumption amount (yuan, 2 decimal places)

balance Balance (yuan, 2 decimal places)

subBalance Subsidy wallet balance (yuan, 2 decimal places)

posModel Consumption mode (1. Fixed value mode 2. Amount mode 3. Key value mode 4. count mode 5. Commodity mode 6. timing mode 7. bookkeeping mode 8. manual consumption 9. Equipment error correction)

hallingCode Restaurant number

hallingName Restaurant name

mealCode Meal division number

mealName Dinner name

devSn Device SN

devAlias Device name

devSerialNum Device serial number

## 2.5 Online Consumption

### 2.5.1 Online Consumption Account Interface

#### 2.5.1.1 Get the Personnel Account Information through the Personnel Number

<b>Post Request URL</b>	http://serverIP:serverPort/api/account/getAccount?personPin=111&access_token={apitoken}
-------------------------	-----------------------------------------------------------------------------------------

<b>Request Mode</b>	GET
<b>Request content</b>	not have
<b>Request Parameter Description</b>	personPin: Personnel number;
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "personPin": "1234568",     "personName": "123",     "personLastName": "",     "accountNo": "1234568",     "balance": 0,     "posMoneyWallet": 0,     "posAllowanceWallet": 0,     "deptCode": "1",     "deptName": "Department Name",     "accountType": 0,     "serialNum": 1,     "consumeSerailnum": 1,     "available": 0,     "accountMember": 0   } }</pre>

	<pre>"accountKindName": "General account" } }</pre>
<b>Response</b> <b>Result</b> <b>Description</b>	<p>personPin Personnel number</p> <p>personName Name of the person</p> <p>personLastName Persons last name</p> <p>accountNo Account number</p> <p>balance Balance (points)</p> <p>posMoneyWallet Cash Wallet balance (points)</p> <p>posAllowanceWallet Subsidy wallet balance (points)</p> <p>deptCode Department number</p> <p>deptName Department name</p> <p>consumeSerailnum Account serial number</p> <p>available Is valid 0 enabled 1 disabled</p> <p>accountMember Is it a member account 0 non-member 1 member</p> <p>accountKindName Account category name</p>

## 2.5.2 Online Consumption and Consumption Record Interface

### 2.5.2.1 Get Consumption Records

<b>Post Request</b> <b>URL</b>	<pre>http://serverIP:serverPort/api/transaction/listPosidTransaction?personPin=1&amp;startDate={ startDate}&amp;endDate={ endDate}&amp;pageNo=1&amp;pageSize=20&amp;access_token={apitoken}</pre>
<b>Request Mode</b>	<pre>GET</pre>

<b>Request content</b>	not have
<b>Request Parameter Description</b>	<p>personPin: Personnel number (not required);</p> <p>startDate: Start time of the query record (not required);</p> <p>endDate: End time of query record (not required);</p> <p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize The number of articles on each page, required, must be greater than 0;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 20,     "total": 1,     "data": [       {         "id": "4028b3b2872b638601872b6e6b630012",         "personPin": "1234568",         "personName": "123",         "personLastName": "",         "accountNo": "1234568",         "typeName": 8,         "money": 1000, </pre>

```
"balance": 9000,  
  
"moneyWallet": 9000,  
  
"allowanceWallet": 0,  
  
"posModel": 8,  
  
"discount": null,  
  
"discountMoney": 0,  
  
"consumeOrder": 0,  
  
"deptCode": "1",  
  
"deptName": "Department Name",  
  
"hallingCode": "1",  
  
"hallingName": "Headquarters",  
  
"mealCode": "1",  
  
"mealName": "Breakfast",  
  
"devSn": "10001",  
  
"accountSerial": 3,  
  
"cardNo": null,  
  
"posTime": "1680019200000",  
  
"createTime": "1680060869474",  
  
"creatorName": "admin",  
  
"logFlag": 2  
}  
],  
  
"offset": 0,
```

	<pre>"lastPage": true } }</pre>
<b>Response Result Description</b>	<p>personPin Personnel number</p> <p>personName Name of the person</p> <p>personLastName Persons last name</p> <p>accountNo Account number</p> <p>typeName Type name (6. Consumption 8. replenishment 9. error correction 10. count 11. bookkeeping 12 offline consumption 13 cancel ordering)</p> <p>money Consumption amount (points)</p> <p>balance Balance (points)</p> <p>moneyWallet Cash Wallet balance (points)</p> <p>allowanceWallet Subsidy wallet balance (points)</p> <p>posModel Consumption mode (1. Fixed value mode 2. Amount mode 3. Key value mode 4. counting mode 5. Commodity mode 6. timing mode 7. bookkeeping mode 8. manual consumption 9. Equipment correction 10. System correction 11 ordering mode 12 intelligent mode)</p> <p>discount Discount</p> <p>discountMoney Discount amount</p> <p>consumeOrder Consumption order</p> <p>deptCode Department number</p> <p>deptName Department name</p> <p>hallingCode Restaurant number</p> <p>hallingName Restaurant name</p>

mealCode	Meal division number
mealName	Dinner name
devSn	Device SN
accountSerial	Account serial number
cardNo	Consumption card number
posTime	Consumption time
createTime	Record the insertion time
creatorName	The Operator
logFlag	Record mark (1. Equipment upload 2. System addition 3. Error correction and complement, 4. wechat 5. alipay 6. offline consumption)

## 2.6 Elevator Control

### 2.6.1 Elevator Device Interface

#### 2.6.1.1 Get Ele Device List [eleDevice/eleList]

<b>Post Request URL</b>	http://serverIP:serverPort/api/eleDevice/eleList?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: page no, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	{ "code": 0, "message": "success", "data": [

	<pre> {   "id": "402856aa6c3bbde1016c3bc2329200ff",   "sn": "0564140100195",   "name": "192.168.218.104",   "type": "EC-100",   "state": "1",   "module": "ele" }, {   "id": "402856aa6c3bbde1016c3bc210d90016",   "sn": "0566141900209",   "name": "192.168.218.11",   "type": "EC-100",   "state": "1",   "module": "ele" } ] } </pre>
<p><b>Response Result Description</b></p>	<p>Data: Device array List;          Id: device id;          Sn: device serial number;          Name: device name;          Type: device model name;          state: state: device state, 1 enabled, 0 disabled;          Module: module, access, attendance, elevator;</p>

### 2.6.1.2 Get Ele Device Info by Sn [eleDevice/getEle/{sn}]

<p><b>Post Request URL</b></p>	<p>http://serverIP:serverPort/api/eleDevice/getEle?sn={sn}&amp;access_token={apitoken}</p>
<p><b>Request Mode</b></p>	<p>GET</p>

<b>Request Content</b>	None
<b>Request Parameter Description</b>	Sn: device sn, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "402856aa6c3bbde1016c3bc2329200ff",     "sn": "5662012052868",     "name": "192.168.214.15",     "type": "EC-100",     "state": "0",     "module": "ele"   } }</pre>
<b>Response Result Description</b>	<p>Data: device object;</p> <p>id: device id;</p> <p>sn: device sn;</p> <p>name: device name;</p> <p>type: device model name;</p> <p>state: state: device state, 1 enabled, 0 disabled;</p> <p>Module: module, access, attendance, elevator;</p>

### 2.6.1.3 Get Ele Device List (With Total Number) [api/v2/eleDevice/eleList]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/eleDevice/eleList?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>pageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0</p>

<p><b>Response Result</b></p>	<pre> {   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 1,     "total": 1,     "data": [{       "id": "8a888e238478c3a6018478ea9cdc0b9c",       "sn": "6013150300019",       "name": "10.8.14.251",       "type": "EC-100",       "status": "1",       "module": "ele"     }],     "offset": 0,     "lastPage": true   } } </pre>
<p><b>Response Result Description</b></p>	<p>data: response result;  Page: current request page number;  Size: data number of current request;  Total: total number;  data: device array list;  Id: device id;  Sn: device sn;  Name: device name;  Type: device model name;  state: device state, 1 enabled, 0 disabled;  Module: module, access, attendance, elevator;</p>

## 2.6.2 Elevator Control Level Interface

### 2.6.2.1 Delete Person Level [eleLevel/deleteLevel]

<b>Post Request URL</b>	http://serverIP:serverPort/api/eleLevel/deleteLevel?pin={pin}&levelIds={levelIds}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Pin: pin, required level Ids: level Ids: level group id List, multiple comma separated, required, string; such as "402856aa6c3bbde1016c3bc2329200ff"
<b>Response Result</b>	Refer to public Response Result
<b>Response Result Description</b>	public response parameter description

### 2.6.2.2 Get Ele Level by Id[eleLevel/getById/{id}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/eleLevel/getById/{id}?access_token={apitoken}
<b>Request Mode</b>	get
<b>Request Content</b>	None
<b>Request Parameter Description</b>	id: level group, required, string; such as: "402856aa6c3bbde1016c3bc2329200ff"
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "402856aa6c3bbde1016c3bc2329200ff",     "name": "public level group"   } }</pre>

	}
<b>Response Result Description</b>	data: level group object; id: level group id; name: level group name;

### 2.6.2.3 Get Ele Level by Name[eleLevel/getByName/{name}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/eleLevel/getByName/{name}?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	name: level group name, required;
<b>Response Result</b>	{ "code": 0, "message": "success", "data": { "id": "402856aa6c3bbde1016c3bc2329200ff", "name": "public level group" } }
<b>Response Result Description</b>	data: level group object; id: level group id; name: level group name;

### 2.6.2.4 Get Ele Level Group List[[eleLevel/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/eleLevel/list?pageNo={pageNo}&page Size={pageSize}&access_token={apitoken}
<b>Request Mode</b>	GET

<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: page, required, must be greater than 0 pageSize: The number of per page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c3bbde1016c3bc2329200ff",       "name": "public level group"     }   ] }</pre>
<b>Response Result Description</b>	data: level group array list; id: level group id; name: level group name;

### 2.6.2.5 Sync Ele Level Group Info to Device[eleLevel/syncLevel]

<b>Post Request URL</b>	http://serverIP:serverPort/api/eleLevel/syncLevel?levelId={levelId}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	levelid: level group, required, string; such as: "402856aa6c3bbde1016c3bc2329200ff"
<b>Response Result</b>	Refer to public response result, other can refer to Appendix-Error code
<b>Response Result Description</b>	public response parameter description

### 2.6.2.6 Sync Person Ele Level [eleLevel/syncPerson]

<b>Post Request URL</b>	http://serverIP:serverPort/api/eleLevel/syncPerson?pin={pin}&levellds={levellds}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Levellds: level group id list, multiple comma separated, required, string; such as: "402856aa6c3bbde1016c3bc2329200ff" pin: pin, required;
<b>Response Result</b>	Refer to public response result, other can refer to Appendix-Error code
<b>Response Result Description</b>	public response parameter description

### 2.6.2.7 Get Elevator access level list(With total number) [/api/v2/eleLevel/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/accLevel/list?pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	PageNo: page no, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	{ "code": 0, "message": "success", "data": { "page": 0, "size": 1, "total": 1, "data": [{

	<pre>                 "id": "8a888e238474d72d018474d8cba8065a",                 "name": "General"             }},             "offset": 0,             "lastPage": true         }     } </pre>
<b>Response Result Description</b>	<p>page: the page number of the current request;</p> <p>size: the amount of data requested for the current page number;</p> <p>total: the total number;</p> <p>data: access levels arrays;</p> <p>id: access level id; such as: "8a888e238474d72d018474d8cba8065a"</p> <p>name: access level name;</p>

## 2.6.3 Elevator Transaction Interface

### 2.6.3.1 Get Elevator Transaction [eleTransaction/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/eleTransaction/list?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>personPin: personPin, [not required]</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p>
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": [ </pre>

```
{
  "id": "402856aa6c1778b3016c1a695c860013",
  "eventTime": "2017-08-04 15:59:12",
  "pin": "",
  "name": "",
  "lastName": "",
  "deptName": "",
  "areaName": "area name",
  "cardNo": "",
  "devSn": "5662012052868",
  "verifyModeName": "others",
  "eventName": "enable device",
  "eventPointName": "",
  "readerName": "others",
  "accZone": null,
  "devName": "192.168.214.15",
  "logId": 11734
},
{
  "id": 402856aa6c1778b3016c1a695c860016,
  "eventTime": "2017-08-04 15:58:49",
  "pin": "",
  "name": "",
  "lastName": "",
  "deptName": "",
  "areaName": "areaname",
  "cardNo": "",
  "devSn": "5662012052868",
  "verifyModeName": "others",
  "eventName": "connect server successfully",
  "eventPointName": "",
  "readerName": "others",
  "devName": "192.168.214.15",
  "logId": 11733
}
```

	<pre> } ] } </pre>
<b>Response Result Description</b>	<p>data: transaction array list;</p> <p>id: 5;</p> <p>eventTime: record device trigger time;</p> <p>pin: pin;</p> <p>name: name;</p> <p>lastName: person name (English);</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p>

### 2.6.3.2 Get Elevator Transaction (With Total Number) [api/v2/eleTransaction/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/eleTransaction/list?personPin={personPin}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>personPin: personPin, [not required]</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater</p>

	than 0, not more than 1000 per page;
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "page": 1,     "size": 2,     "total": 2,     "data": [{       "id": "402856 aa6c1778b3016c1a695c860013",       "eventTime": "2017-08-04 15:59:12",       "pin": "",       "name": "",       "lastName": "",       "deptName": "",       "areaName": "Area",       "cardNo": "",       "devSn": "5662012052868",       "verifyModeName": "Other",       "eventName": "Device Started",       "eventPointName": "",       "readerName": "Other",       "accZone": null,       "devName": "192.168.214.15",       "logId": 11734     },     {       "id": "402856 aa6c1778b3016c1a695c860016",       "eventTime": "2017-08-04 15:58:49",       "pin": "",       "name": "",       "lastName": "",       "deptName": "", </pre>

	<pre> "areaName": "Area", "cardNo": "", "devSn": "5662012052868", "verifyModeName": "Other", "eventName": "Connect to Server", "eventPointName": "", "readerName": "Other", "devName": "192.168.214.15", "logId": 11733     }  ], "offset": 1, "lastPage": true     } } </pre>
<p><b>Response Result Description</b></p>	<p>data: result</p> <p>page: the page number of the current request;</p> <p>size: the number of data requested by the current page number;</p> <p>total: total number;</p> <p>data: event record array collection;</p> <p>id: "402856aa6c1778b3016c1a695c860016",</p> <p>eventTime: recording device trigger time;</p> <p>pin: personnel number;</p> <p>name: name of the personnel;</p> <p>lastName: the name of the personnel, only useful in English;</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card number;</p> <p>devSn: device serial number;</p> <p>verifyModeName: verify mode name;</p> <p>eventName: event description;</p> <p>eventPointName: event departure point name;</p>

	<p>readerName: name of the reader head;</p> <p>devName: device name;</p> <p>logId: event index value, only have value according to whether the device supports it or not;</p>
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 2.6.4 Floor Interface

### 2.6.4.1 Get Floor Info List[floor/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/floor/list?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: page, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c08e7e4016c092400580021",       "floorNo": 1,       "name": "192.168.214.15-1",       "deviceId": "402856aa6c3c5063016c3cb3a0360005",     },     {       "id": "402856aa6c08e7e4016c090be94a000a",       "floorNo": 2,       "name": "192.168.214.15-2",       "deviceId": "402856aa6c3c5063016c3cb3a0360005"     },     {       "id": "402856aa6c083827016c08582f8f0012",</pre>

	<pre> "floorNo":3, "name": "192.168.214.15-3", "deviceId": "402856aa6c3c5063016c3cb3a0360005", }, { "id": "402856aa6c3baabb016c3bb37365000a", "floorNo":4, "name": "192.168.214.15-4", "deviceId": "402856aa6c3c5063016c3cb3a0360005", } ] } </pre>
<b>Response Result Description</b>	<p>data: floor array list;</p> <p>id: floor id;</p> <p>floorNo: floor no;</p> <p>name: floor name;</p> <p>deviceId: device id;</p>

### 2.6.4.2 Remote Close by Floor Id[floor/remoteCloseById]

<b>Post Request URL</b>	http://serverIP:serverPort/api/floor/remotecloseById?floorId={floorId}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	floorId: floorid, required, string; such as: "402856aa6c3c5063016c3cb3a0360005"
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.6.4.3 Remote Open by Floor Id[floor/remoteOpenById]

<b>Post Request URL</b>	http://serverIP:serverPort/api/floor/remoteOpenById?floorId={floorId}&interval={interval}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	floorId: floor id, required, string; such as: "402856aa6c3c5063016c3cb3a0360005" interval: door opening time, seconds, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.6.4.4 Get Floor Info List (With Total Number) [api/v2/floor/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/floor/list?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	PageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 3,     "total": 10,     "data": [       {</pre>

	<pre>                 "id": "8a888e238478c3a6018478ea9cf40bc1",                 "floorNo": 1,                 "name": "10.8.14.251(1)",                 "deviceId": "8a888e238478c3a6018478ea9cdc0b9c"             },             {                 "id": "8a888e238478c3a6018478ea9cf40bc2",                 "floorNo": 2,                 "name": "10.8.14.251(2)",                 "deviceId": "8a888e238478c3a6018478ea9cdc0b9c"             },             {                 "id": "8a888e238478c3a6018478ea9cf40bc3",                 "floorNo": 3,                 "name": "10.8.14.251(3)",                 "deviceId": "8a888e238478c3a6018478ea9cdc0b9c"             }         ],         "offset": 0,         "lastPage": false     } } </pre>
<p><b>Response Result Description</b></p>	<p>data: result;                  Page: current request page;                  Size: current data number of current request page;                  Total: total number;                  data: floor array list;                  id: floor id;                  floorNo: floor no;                  name: floor name;                  deviceId: device id;</p>

## 2.7 Visitor Management

### 2.7.1 Visitor Registration Check Out Interface

#### 2.7.1.1 Add Visitor Registration[visRegistration/add]

<b>Post Request URL</b>	http://serverIP:serverPort/api/visRegistration/add?&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "persPersonPin": "1",   "certType": "8",   "certNum": "456888",   "visEmpName": "max",   "visitEmpPhone": "",   "company": "",   "visitReason": "visit",   "visitorCount": "1",   "startTime": "2018-08-31 14:40:00",   "endTime": "2018-08-31 23:59:59",   "cardNo": "",   "visLevels": "" }</pre>
<b>Request Parameter Description</b>	<p>"persPersonPin": "Interviewee pin", required  certType: certificate type, required; details:</p> <pre>{   2: national ID card; 3: Passport; 4: Driver's license; 5: Work Permit;   6: Real-name card; 7: driving license; 8: others; }</pre> <p>"certNum": visitor certificate no, required  "visitEmpName": visitor name, required</p>

	<p>"visitEmpPhone": visitor mobile phone,  "company": company,  "visitReason": visit reason, required  visitorCount: the number of visitors, required  startTime: start time, yyyy-MM-dd mm:hh:ss, required  endTime: end time, yyyy-MM-dd mm:hh:ss, required  cardNo: card no  visLevels: visitor level group id, string; such as:  402856aa6bff4b1a016bff5374800c0e,402856aa6c3c5063016c3c  b3a0360005</p>
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.7.1.2 Visitor Check Out [visRegistration/exit]

<b>Post Request URL</b>	http://serverIP:serverPort/api/visRegistration/exit?certType={certType}&certNum={certNum}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>certType:certificate type, required; details:</p> <pre>{     2: national ID card; 3: Passport; 4: Driver's license; 5: Work Permit;     6: real-name card; 7: driving license; 8: others; }</pre> <p>"certNum": visitor certificate no, required</p>
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.7.1.3 Get Visitor Dynamic QR code

<b>Post Request URL</b>	http://serverIP:serverPort/api/visRegistration/getQrCode/{pin}?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	{ "pin": "80000001"
<b>Request Parameter Description</b>	Pin: visitor number, required
<b>Response Result</b>	{ "code": 0, "data": {}, "message": "string" }
<b>Response Result Description</b>	Data: visitor's dynamic qr code, base 64 data;

### 2.7.1.4 Get Visitor Dynamic QR code--V2

<b>Post Request URL</b>	http://serverIP:serverPort/api/visRegistration/getQrCode?pin={pin}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	{ "pin": "80000001"
<b>Request Parameter Description</b>	Pin: visitor id, required
<b>Response Result</b>	{ "code": 0, "data": {}, "message": "string" "data":

	"2#6SQLIaSLhprGhpkCgFrhHfdyZXFJxK2DLy+oLVolmol="
	}
<b>Response Result Description</b>	Data: visitor's dynamic qr code, base 64 data;

### 2.7.1.5 Retrieve a Dynamic QR Code Based on ID/Phone Number

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/visRegistration/getQrCodeByCertNumber?access_token={apitoken}&certNumber={certNumber}&certType={certType}&phone={phone}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	certNumber: Certification ID, required certType: Certification Type Phone: Mobile phone
<b>Response Result</b>	{ "code": 0, "data": {}, "message": "string" "data": "2#6SQLIaSLhprGhpkCgFrhHfdyZXFJxK2DLy+oLVolmol="
	}
<b>Response Result Description</b>	Data: visitor's dynamic QR code, base 64 data;

### 2.7.1.6 Retrieve Visitor List

<b>Post Request URL</b>	http://serverIP:serverPort/api/visVisitor/getVisitorList?access_token={apitoken}&certNumbers={certNumbers}&names={names}&pageNo={pageNo}&pageSize={pageSize}
-------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize: Page size, required, must be greater than 0, No more than 1000 items per page.</p> <p>certNumbers: Certification ID.</p> <p>names: Visitor Name</p>
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": [     {       "id": "8a888ea18d584d12018d592429020014",       "pin": "800000001",       "name": "123",       "lastName": null,       "certType": "8",       "certNumber": "123",       "company": "",       "carPlate": "",       "phone": "",       "email": "",       "status": null, </pre>

	<pre>                 "enabled": true,                 "nationality": null,                 "gender": ""             }         ]     </pre>
<b>Response Result Description</b>	<p>id: Visitor information ID</p> <p>pin: Visitor number</p> <p>name: Visitor's first name</p> <p>lastName: Visitor's last name</p> <p>certType: ID type</p> <p>certNumber: ID number</p> <p>company: Company</p> <p>carPlate: License plate</p> <p>phone: Mobile phone number</p> <p>email: Email</p> <p>status: Status</p> <p>enabled: Enable/disable</p> <p>nationality: Nationality</p> <p>gender: Gender</p>

### 2.7.1.7 Obtain the list of visitor history

[records\[v2/visRegistration/getTransactionList\]](#)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/visRegistration/getTransactionList?access_token={apitoken}&certNum={certNum}&phone={phone}&pageNo={pageNo}&pageSize={pageSize}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: Page number, required, must be greater than 0;

	<p>pageSize: Page size, required, must be greater than 0, No more than 1000 items per page.</p> <p>certNumbers: Certification ID.</p> <p>names: Visitor Name</p>
<p><b>Response Result</b></p>	<pre> {   "code": 0,   "message": "success",   "data": [     {       "id": "8a8890a19810d399019811052aa20023",       "persPersonPin": "1",       "certType": 8,       "certNum": "71601",       "visEmpName": "The name of 71601",       "visitEmpPhone": "",       "company": "",       "visitReason": "",       "visitorCount": 1,       "startTime": "2025-07-16 10:16:48",       "endTime": null,       "cardNo": "",       "visLevels": null,       "facePhoto": null,       "certPhoto": null,     }   ] } </pre>

	<pre> "signaturePhoto": null,  "enterPlaceId": null,  "email": "",  "visEmpLastName": "surname",  "carPlate": "",  "enterGoodsPhoto": null,  "deptCode": null,  "visEmpPin": "800000007",  "visitedEmpDept": "Department Name"      }    ]  } </pre>
<p><b>Response Result Description</b></p>	<p>id: Visitor record ID  persPersonPin: Visitee number  certType: ID type  certNum: ID number  visEmpPin: Visitor number  visEmpName: Visitor's name  visEmpLastName: Visitor's last name  visitEmpPhone: Visitor's mobile phone number  company: Company  visitReason: Purpose of visit  visitorCount: Number of visitors  startTime: Entry time  endTime: Departure time  cardNo: Card number  carPlate: License plate  email: Email</p>

visitedEmpDept: Name of the visited department

## 2.7.2 Visitor Reservation Interface

### 2.7.2.1 Add Visitor Reservation[visReservation/add]

<b>Post Request URL</b>	http://serverIP:serverPort/api/visReservation/add?access_token={a pitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "persPersonPin": "5",   "certType": "3",   "certNum": "4512",   "visitEmpName": "Mary",   "visitEmpPhone": "13015648651",   "company": "ZKTeco",   "visitReason": "visit",   "visitDate": "2018-08-31 14:40:00"   "visitEndDate": "2018-08-31 23:59:59"   "email": "123@outlook.com",   "carPlate": "7P8 Q2X" }</pre>
<b>Request Parameter Description</b>	<p>persPersonPin: interviewee pin, required;</p> <p>certType: certificate type, required; details:</p> <pre>{   2:national ID card; 3:Passport; 4:Driver's license; 5:Work Permit;   6:Real-name card; 7:driving license; 8:others; }</pre> <p>certNum: certificate no, required;</p> <p>visitEmpName: interviewee name, required;</p> <p>visitEmpPhone: visitor phone;</p> <p>company: company;</p>

	visitReason: visit reason, required; visitDate: visit date, yyyy-MM-dd mm:hh:ss, required; visitEndDate: visit date, yyyy-MM-dd mm:hh:ss, required;
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.7.2.2 Delete Visitor Reservation[visReservation/del]

<b>Post Request URL</b>	http://serverIP:serverPort/api/visReservation/del?certType={certType}&certNum={certNum}&visitDate={visitDate}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	certType: certificate type, required; details: { 2: national ID card; 3: Passport; 4: Driver's license; 5:Work Permit; 6: Real-name card; 7: driving license; 8:others; } certNum: certificate no, required; visitDate: visit date, required; [The system will find out the only visitor reservation information and delete it according to the certificate type, ID number, and visit date]
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

## 2.7.3 Visitor Level Interface

### 2.7.3.1 Get Visitor Level Group List[visLevel/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/visLevel/list?pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	PageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6bff4b1a016bff5374800c0e",       "name": "public level group"     }   ] }</pre>
<b>Response Result Description</b>	data: level group array list; id: level group id; name: level groupname;

## 2.8 Parking

### 2.8.1 Parking Basic Management Interface

#### 2.8.1.1 Retrieve the Device Channel List

<b>Post Request</b>	http://serverIP:serverPort/api/v1/parkBase/getParkChannelList?pageNo=1&pageSiz
---------------------	--------------------------------------------------------------------------------

<b>URL</b>	e=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: Page, required, Must be grater than 0; pageSize:page size, required, Must be grater than 0;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "data": [       {         "channelName": "Exit",         "channelId": "4028b59c82101c36018210a7aabb05a0"       },       {         "channelName": "Entrance",         "channelId": "4028b59c81faa6120181fb2207e803bc"       }     ],     "totalCount": 2   } }</pre>
<b>Response Result Description</b>	data: Device array collection; id: Device channel ID; name: Device channel name.

### 2.8.1.2 The Specified Device Channel is Opened

<b>Post Request URL</b>	http://serverIP:serverPort/api/v1/parkBase/openGateChannel?channelId=4028b59c81faa6120181fb2207e803bc&access_token={apitoken}
-------------------------	-------------------------------------------------------------------------------------------------------------------------------

<b>Request Mode</b>	POST
<b>Request Content</b>	not have
<b>Request Parameter Description</b>	channelId: Channel id, required; e. g., "4028b59c81faa6120181fb2207e803bc"
<b>Response Result</b>	See the Universal Response Results
<b>Response Result Description</b>	Description of the generic response parameters

### 2.8.1.3 List of Parking Yard Areas (v2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/parkBase/getParkingAreaList?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	NONE
<b>Request Parameter Description</b>	pageNo: Page number, required, must be greater than 0; pageSize The number of articles on each page, required, must be greater than 0;
<b>Response Results</b>	{ "code": 0, "message": "success", "data": { "page": 0, "size": 10,

	<pre> "total": 1,   "data": [     {       "id": "4028b59c8d3a34be018d3a5d1b2e0009",       "name": "B1",       "parentId": ""     }   ],   "offset": 0,   "lastPage": true } </pre>
<b>Response Result Description</b>	<p>Data: array sets;</p> <p>id: parking area id;</p> <p>parentId: parent parking area id;</p> <p>name: parking area name;</p>

#### 2.8.1.4 List of Parking Vehicle Types (v2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/parkBase/getParkingCarTypeList?pageNo=1&page Size=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	None

<b>Request Parameter Description</b>	<p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize The number of articles on each page, required, must be greater than 0;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 10,     "total": 1,     "data": [       {         "id": "4028b59c8d3a34be018d3a5d1b2e0009",         "Name": "Small car A"       }     ],     "offset": 0,     "lastPage": true   } }</pre>
<b>Response Result Description</b>	<p>Data: array sets;</p> <p>id: type id;</p> <p>name: name;</p>

### 2.8.1.5 Get Parking Discount Strategy List (v2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/parkBase/getParkingDiscountList?pageNo=1&page Size=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	not have
<b>Request Parameter Description</b>	<p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize The number of articles on each page, required, must be greater than 0;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 10,     "total": 3,     "data": [       {         "discountName": "minutes",         "discountType": 3,         "discountId": "4028b59c919c3c4001919ccdf898000c",         "content": "120"       },       {</pre>

	<pre> "discountName": "Amount", "discountType": 2, "discountId": "4028b59c919c3c4001919ccda746000a", "content": "10.0" }, { "discountName": "Free at all", "discountType": 1, "discountId": "4028b59c919c3c4001919cccfa010007", "content": "0" } ], "offset": 0, "lastPage": true } } </pre>
<p><b>Response Result Description</b></p>	<p>Data: array sets;</p> <p>discountId: Discount id;</p> <p>discountType: 1, discount type</p> <p>discountName: Name;</p> <p>Content: Discount content</p>

### 2.8.1.6 Parking Entry and Exit List (v2)

<p><b>Post Request</b></p>	<p><a href="http://serverIP:serverPort/api/v2/parkBase/getParkingEntranceList?pageNo=1&amp;page">http://serverIP:serverPort/api/v2/parkBase/getParkingEntranceList?pageNo=1&amp;page</a></p>
----------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>URL</b>	Size=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	not have
<b>Request Parameter Description</b>	<p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize The number of articles on each page, required, must be greater than 0;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 10,     "total": 1,     "data": [       {         "id": "4028b59c89d90a5e0189d987deac002f",         "name": "A-E1",         "areald": "4028b59c89bf49dd0189bf50ceb7004e",       }     ],     "offset": 0,     "lastPage": true   } }</pre>

	<pre> }</pre>
<b>Response Result Description</b>	Data: array sets; id: Regional import and export id; name: name; areald: region id;

### 2.8.1.7 Parking Space List (v2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/parkBase/getParkingSpaceList?pageNo=1&pageSiz e=20&areald=4028b59a788258f501788262e5d209ed&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	None
<b>Request Parameter Description</b>	pageNo: Page number, required, must be greater than 0; pageSize The number of articles on each page, required, must be greater than 0; areald: Area, required to fill, area id;
<b>Response Results</b>	<pre> { "code": 0, "message": "success", "data": { "page": 0, "size": 10, "total": 1, "data": [ { </pre>

	<pre>                 "id": "4028b59c919cd64201919cddab770106",                 "name": "K005",                 "areald": "4028b59a788258f501788262e5d209ed"             }         ],         "offset": 0,         "lastPage": true     } } </pre>
<b>Response Result Description</b>	Data: array sets; id: parking space id; Name: Parking Space name; areald: parking area id;

### 2.8.1.8 List of Parking Channels (v2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/parkBase/getParkingChannelList?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	None
<b>Request Parameter Description</b>	pageNo: Page number, required, must be greater than 0; pageSize The number of articles on each page, required, must be greater than 0;
<b>Response</b>	{

<b>Results</b>	<pre> "code": 0, "message": "success", "data": { "page": 0, "size": 10, "total": 1, "data": [ { "channelName": "Export", "channelId": "4028b59c82101c36018210a7aabb05a0" } ], "offset": 0, "lastPage": true } </pre>
<b>Response Result Description</b>	<p>Data: array sets;</p> <p>channelId: parking channel id;</p> <p>channelName: Parking channel name;</p>

## 2.8.2 Park Authorization Interface

### 2.8.2.1 Authorize Park Black and White List [parkAuthorize/addBlackWhite]

<b>Post Request URL</b>	<a href="http://serverIP:serverPort/api/parkAuthorize/addBlackWhite?type={t">http://serverIP:serverPort/api/parkAuthorize/addBlackWhite?type={t</a>
-------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------

	ype}&carNumber={carNumber}&fromTime={fromTime}&toTime={toTime}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	type: Black and White List type, 0: White List, 1:Black list carNumber: license plate number, such as: A32212 fromTime: start time, 2018-08-17 toTime: end time, 2019-08-16
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.8.2.2 Delete Park Black and White List[parkAuthorize/delBlackWhite]

<b>Post Request URL</b>	http://serverIP:serverPort/api/parkAuthorize/delBlackWhite?carNumber={carNumber}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	carNumber: license plate number, such as: A32212
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.8.2.3 Edit Park Black and White List[parkAuthorize/editBlackWhite]

<b>Post Request URL</b>	http://serverIP:serverPort/api/parkAuthorize/editBlackWhite?type={type}&carNumber={carNumber}&oldCarNumber={oldCarNumber}&fromTime={fromTime}&toTime={toTime}&access_token={apitoken}
-------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>type: Black and White List type, 0: White List, 1:Black list</p> <p>carNumber: new license plate number, such as A32212</p> <p>oldCarNumber: old license plate number, such as A32212</p> <p>fromTime: start time, 2018-08-17(White List need to be filled in)</p> <p>toTime: end time 2019-08-16(White List need to be filled in)</p>
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

#### 2.8.2.4 Edit the Parking Allowed / Prohibited List (v2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/parkPlate/addAllowDisabled?type={ type}&carNumber={ carNumber}&oldCarNumber={oldCarNumber}&parkCode={parkCode}&fromTime={ fromTime}&toTime={ toTime}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	not have
<b>Request Parameter Description</b>	<p>Type: allowed / prohibited list type, 0: allowed list, 1: prohibited list</p> <p>carNumber: New license plate number, such as: Min A32212</p> <p>oldCarNumber: Old license plate number, such as: not required</p> <p>parkCode: Car id, such as: not required</p> <p>fromTime: Start time, 2018-08-17</p> <p>toTime: End time 2019-08-16</p>
<b>Response</b>	See General Response Results and others see Appendix-Exception Dictionary

<b>Results</b>	
<b>Response Result Description</b>	Description of the generic response parameters

### 2.8.2.5 New and Edit Parking Personnel License Plates

<b>Post Request URL</b>	http://serverIP:serverPort/api/v1/parkPlate/addPlate?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "pin": "1234",   "carNumber": "Min A32212",   "oldCarNumber": "" }</pre>
<b>Request Parameter Description</b>	<p>carNumber: New license plate number, such as: Min A32212</p> <p>oldCarNumber: Old license plate number, can be empty, such as: Min A32212</p> <p>"pin": personnel pin, such as: 12345</p>
<b>Response Result</b>	See general Response Results and others see Appendix-Exception Dictionary
<b>Response Result Description</b>	Description of the generic response parameters

### 2.8.2.6 Delete the Parking Persons License Plates

<b>Post Request URL</b>	http://serverIP:serverPort/api/v1/parkPlate/delPlate?access_token={apitoken}
-------------------------	------------------------------------------------------------------------------

<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "pin": "1234",   "carNumber": "Min A32212" }</pre>
<b>Request Parameter Description</b>	<p>carNumber: New license plate number, such as: Min A32212</p> <p>"pin": personnel pin, such as: 12345</p>
<b>Response Result</b>	See general Response Results and others see Appendix-Exception Dictionary
<b>Response Result Description</b>	Description of the generic response parameters

### 2.8.2.7 Parking of Authorized Vehicle List (v2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/parkPlate/addFixedVehicle?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	<pre>{   "amount": "0",   "areald": "Area Id",   "carNumber": "Liao A12345",   "carTypeid": "CarType Id",   "endTime": "YYYY-MM-DD",   "entrancelds": "1,2,3", }</pre>

	<pre>"pin": "Person Pin", "spaceId": "Space Id", "startTime": "YYYY-MM-DD" }</pre>
<b>Request Parameter Description</b>	<pre>"amount": "Amount", "areaId": "Area Id", "carNumber": "Liao A12345", "carTypeId": "CarType Id", "endTime": "YYYY-MM-DD", "entranceIds": "1,2,3", "pin": "Person Pin", "spaceId": "Space Id", not required "startTime": "YYYY-MM-DD"</pre>
<b>Response Results</b>	<pre>{ "code": 0, "message": "success", "data": null }</pre>
<b>Response Result Description</b>	Description of the generic response parameters

### 2.8.2.8 Parking Delete Authorized Vehicle List (v2)

<b>Post Request</b>	<a href="http://serverIP:serverPort/api/v2/parkPlate/delFixedVehicle?access_token={apitoken}">http://serverIP:serverPort/api/v2/parkPlate/delFixedVehicle?access_token={apitoken}</a>
---------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>URL</b>	n}
<b>Request Mode</b>	POST
<b>Request content</b>	{ "carNumber": "Liao A12345", "pin": "Person Pin" }
<b>Request Parameter Description</b>	"carNumber": "A12345", "pin": "Person Pin"
<b>Response Results</b>	{ "code": 0, "message": "success", "data": null }
<b>Response Result Description</b>	Description of the generic response parameters

### 2.8.2.9 Parking Query for Fixed Vehicles (v2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/parkPlate/getFixedVehicle?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	{ "carNumber": "Liao A12345",

	<pre>"pin": "Person Pin" } </pre>
<b>Request Parameter Description</b>	<pre>"carNumber": "A12345", "pin": "Person Pin" </pre>
<b>Response Results</b>	<pre>{ "code": 0, "message": "success", "data": null } </pre>
<b>Response Result Description</b>	<pre>data: record "carNumber": license plate number "pin": Person Pin "auth": "auth" is authorized, "noAuth" is unauthorized </pre>

## 2.8.3 Park Transaction Interface

### 2.8.3.1 Get Park Entry Transaction [parkTransaction/listParkRecordin]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/parkTransaction/listParkRecordin?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<pre>carNumber: carNumber, [not required] startDate: startDate, [not required] endDate: endDate, [not required] </pre>

	<p>PageNo: page no, required, the value must be greater than 0;          pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p>
<p><b>Response Result</b></p>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c1778b3016c1a695c860016",       "parkAreaName": " park area",       "name": " driver name",       "carNumber": "car licence plate number",       "carTypeName": "car type",       "channelName": "channel name",       "channelState": " in and out state",       "eventType": "transaction type",       "checkInTime": "check-in time"     },     {       "id": "402856aa6c1778b3016c1a695c860018",       "parkAreaName": "park area",       "name": " driver name",       "carNumber": "car licence plate number",       "carTypeName": "car type",       "channelName": "channel name",       "channelState": "in and out state",       "eventType": "transactiontype",       "checkInTime": "check-in time"     }   ] }</pre>
<p><b>Response Result Description</b></p>	<p>data: transaction array list;          id: "402856aa6c1778b3016c1a695c860018",</p>

	<p>parkAreaName: park area;</p> <p>name: driver name;</p> <p>carNumber: car licence plate number;</p> <p>carTypeName: car type;</p> <p>channelName: channel name;</p> <p>channelState: in and out state;</p> <p>eventType: transaction type;</p> <p>checkInTime: check-in time</p>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.8.3.2 Get Park Exit Transaction[parkTransaction/listParkRecordout]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/parkTransaction/listParkRecordout?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>carNumber: carNumber, [not required]</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, the value must be greater than 0, not more than 1000 per page;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c1778b3016c1a695c860018",       "parkAreaName": "park area",       "name": "driver name",       "carNumber": "car licence plate number",       "carTypeName": " car type",       "channelName": "channel name", </pre>

	<pre> "channelState": "in and out state", "eventType": "transaction type", "checkOutTime": "check-out time" }, { "id": "402856aa6c1e1def016c1e2073e80001", "parkAreaName": "park area", "name": "driver name", "carNumber": "car licence plate number", "carTypeName": "car type", "channelName": "channel name", "channelState": "in and out state", "eventType": "transaction type", "checkOutTime": "check-out time" } ] } </pre>
<p><b>Response Result Description</b></p>	<p>data: transaction array list;  id: "402856aa6c1e1def016c1e2073e80001",  parkAreaName: park area;  name: driver name;  carNumber: car licence plate number;  carTypeName: car type;  channelName: channel name;  channelState: in and out state;  eventType: transaction type;  checkOutTime: check-out time</p>

### 2.8.3.3 Vehicle Admission List (v2)

<p><b>Post Request</b></p>	<p>http://serverIP:serverPort/api/v2/parkTransaction/listParkRecordin?carNumber={c</p>
<p><b>URL</b></p>	<p>arNumber}&amp;startDate={startDate}&amp;endDate={endDate}&amp;pageNo=1&amp;pageSize=20&amp;a</p>

	ccess_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	not have
<b>Request Parameter Description</b>	<p>carNumber: License plate number, not required</p> <p>startDate: Start time, not required, together with the end time ("yyyy-MM-dd HH:mm:ss")</p> <p>endDate: End time, not required, together with the start time ("yyyy-MM-dd HH:mm:ss")</p> <p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize: The number of each page, required, must be greater than 0, each page shall not exceed 1000;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 1,     "total": 1093,     "data": [       {         "id": "4028b59c919255c40191928f905000f4",         "parkingAreaName": "Development Zone",         "name": ""       }     ]   } }</pre>

	<pre> "carNumber": "Liao F4J227", "carTypeName": "Temporary Mini Car A", "channelName": "SVZT-175", "channelState": "1", "eventType": "1", "checkInTime": "2024-08-27 13:53:30.0" } ], "offset": 0, "lastPage": false } } </pre>
<p><b>Response Result Description</b></p>	<p>id: "402856aa6c1778b3016c1a695c860018",</p> <p>parkingAreaName: Parking area;</p> <p>Name: Name of the car owner,;</p> <p>carNumber: License plate number;</p> <p>carTypeName: Car type;</p> <p>channelName: Channel name;</p> <p>channelState: Import and export status; (1: large entry, 2: large exit, 3: small entry, 4: small exit, 5: central charge fixed point, 6: central charge export)</p> <p>eventType: Event type; 1: general card record; 2: fixed card to temporary card; 3: lost card appearance; 4: overtime charge record; 5: manual opening record (through controller or software); 6: cancel opening; 7: monthly card charge; 8: visitor VIP car; 9: license plate correction; 10: unmatched record; 11: online prepayment (wechat</p>

payment); 12: paid (wechat payment)

checkInTime: Admission time

### 2.8.3.4 Vehicle Exit List (v2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/parkTransaction/listParkRecordout?carNumber={carNumber}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	not have
<b>Request Parameter Description</b>	<p>carNumber: License plate number, not required</p> <p>startDate: Start time, not required, together with the end time ("yyyy-MM-dd HH:mm:ss")</p> <p>endDate: End time, not required, together with the start time ("yyyy-MM-dd HH:mm:ss")</p> <p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize: The number of each page, must be greater than 0, each page shall not exceed 1000;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 1,     "total": 1856, </pre>

	<pre> "data": [   {     "id": "4028b59c919255c4019196d939b7110d",     "parkingAreaName": "Development Zone",     "name": "",     "carNumber": "Cloud VD 0 D 0 learning",     "carTypeName": "Temporary Mini Car A",     "channelName": "SVZT-202",     "channelState": "2",     "eventType": "1",     "checkOutTime": "2024-08-28 10:37:41.0",     "parkingTime": "-"   } ], "offset": 0, "lastPage": false } </pre>
<p><b>Response Result Description</b></p>	<p>Data: array sets;</p> <p>id: "402856aa6c1e1def016c1e2073e80001";</p> <p>parkingAreaName: Parking area;</p> <p>Name: Name of the car owner,;</p> <p>carNumber: License plate number;</p>

carTypeName: Car type;
channelName: Channel name;
channelState: Import and export status; (1: large entry, 2: large exit, 3: small entry, 4: small exit, 5: central charge fixed point, 6: central charge export)
eventType: Event type; 1: general card record; 2: fixed card to temporary card; 3: lost card appearance; 4: overtime charge record; 5: manual opening record (through controller or software); 6: cancel opening; 7: monthly card charge; 8: visitor VIP car; 9: license plate correction; 10: unmatched record; 11: online prepayment (wechat payment); 12: paid (wechat payment)
checkOutTime: Appearance time

## 2.8.4 Parking Cost Interface

### 2.8.4.1 Parking Vehicle Billing

<b>Post Request URL</b>	http://serverIP:serverPort/api/v1/parkCost/vehicleBilling&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	{ "carNumber": "辽 A12345" }
<b>Request Parameter Description</b>	carNumber: license plate;
<b>Response Results</b>	{ "code": 0,

	<pre> "message": "success",  "data": {    "carNumber": "辽 A12345",    "recordId": "4028b54595dc76320195eb1e2fba0836",    "carAuth": "temp",    "finalAmount": "233.8",    "chargeType": "tempCharge",    "checkInTime": "2025-03-31 15:32:58",    "parkingTime": "1 02:43:42"  }  } </pre>
<p><b>Response Result Description</b></p>	<p>carNumber: license plate;</p> <p>recordId: entrance sign;</p> <p>carAuth: temp;</p> <p>finalAmount: fee amount;</p> <p>chargeType: tempCharge;</p> <p>checkInTime: admission Time;</p> <p>parkingTime: parking duration;</p>

### 2.8.4.2 Parking Vehicle Payment Result Notification

<p><b>Post Request URL</b></p>	<p>http://serverIP:serverPort/api/v1/parkCost/payNotify&amp;access_token={apitoken}</p>
<p><b>Request Mode</b></p>	<p>POST</p>

<b>Request content</b>	<pre>{   "carNumber": "辽 A12345",   "paymentType": "string",   "recordId": "string" }</pre>
<b>Request Parameter Description</b>	<p>carNumber: license plate;          paymentType: payment method          recordId: entrance sign;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "carNumber": "辽 A12345",     "recordId": "string",     "paymentType": "string"   } }</pre>
<b>Response Result Description</b>	<p>carNumber: license plate;          recordId: entrance sign;          paymentType: payment method</p>

## 2.9 Entrance Control

### 2.9.1 Entrance Control Device Interface

#### 2.9.1.1 Get Entrance Control Device List [api/psgdDvice/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgdDvice/list?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: pageno, required, the value must be greater than 0; pageSize: the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "8a888e8f820060eb018200e487a50a67",       "sn": "0564140100195",       "name": "192.168.218.104",       "type": "PGIC10",       "state": "1",       "module": "psg",       "gateType": "0"     }   ] }</pre>
<b>Response Result Description</b>	Data: device array list; Id: device id; Sn: device sn; Name: device name;

	<p>state: device state, 1 enabled, 0 disabled;</p> <p>Module: module, access, attendance, elevator;</p> <p>Type: device type name;</p> <p>gateType: Entrance Control device type, 0-unknow(device is not connected to entrance control terminals); 1-Tripod Turnstile; 2-Flap Barrier; 3-Swing Barrier</p>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.9.1.2 Get Entrance Control Device Info by SN [/api/psgDevice/getBySn]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgDevice/getBySn?sn={sn}&access_to ken={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Sn: device sn, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "8a888e8f820060eb018200e487a50a67",     "sn": "CKN4213360081",     "name": "10.8.14.234",     "type": "InBio-PGates",     "status": "1",     "module": "psg",     "gateType": "0"   } }</pre>
<b>Response Result Description</b>	<p>Data: device object;</p> <p>Id: device id;</p> <p>Sn: device sn;</p> <p>Name: device name;</p>

	Type: device model type; state: device state, 1 enabled, 0 disabled; Module: module, psg: entrance control module; gateType: Entrance Control device type, 0-unknow(device is not connected to entrance control terminals); 1-Tripod Turnstile; 2-Flap Barrier; 3-Swing Barrier
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.9.1.3 Get Entrance Control Device List (With Total Number)

[api/v2/psgDevice/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/psgDevice/list?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: pageno, required, the value must be greater than 0; pageSize: the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 1,     "total": 1,     "data": [{       "id": "8a888e238488391e018488703e4b0012",       "sn": "CKN3214660034",       "name": "10.8.14.249",       "type": "InBio-PGates",       "status": "1",       "module": "psg",       "gateType": "0"     }],     "offset": 0,   } } </pre>

	<pre> "lastPage": true     }   } </pre>
<b>Response Result Description</b>	<p>data: the returned result;</p> <p>page: the current page number requested;</p> <p>size: the number of data requested by the current page number;</p> <p>total: the total number;</p> <p>Data: device array list;</p> <p>Id: device id;</p> <p>Sn: device sn;</p> <p>Name: device name;</p> <p>state: device state, 1 enabled, 0 disabled;</p> <p>Module: module, access, attendance, elevator;</p> <p>Type: device type name;</p> <p>gateType: Entrance Control device type, 0-unknown(device is not connected to entrance control terminals); 1-Tripod Turnstile; 2-Flap Barrier; 3-Swing Barrier</p>

## 2.9.2 Entrance Control Gate Interface

### 2.9.2.1 Get the State of All Gates [psgGate/allGateState]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgGate/allGateState?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	None
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": [ </pre>

	<pre> {   "id": "8a888e8f820060eb018200e488650abb",   "name": "10.8.14.234-1",   "deviceId": "8a888e8f820060eb018200e487a50a67",   "connect": "1",   "gateState": "0",   "alarm": "0",   "relay": "0" } ] } </pre>
<p><b>Response Result Description</b></p>	<p>Data: entrance control gate state array List;</p> <p>Id: gate id;</p> <p>Name: entrance control device name;</p> <p>DeviceId: device id;</p> <p>Connection: connection state, 0 offline, 1online, 2 disabled;</p> <p>gateState: gate state (0 offline-device is not connect to entrance control gate, default: 0, 1 Online, 2 Disabled, 3 Lockdown)</p> <p>Alarm: alarm description (0-none, 1-be opened accidentally, 2-tampering, 3-duress password opening,4- duress fingerprint opening, 5-duress opening,6-Infraed ray exception, 7=gate error, None-unknown)</p> <p>relay: Relay state (0-off, 1-Exit open,2-Entry Open, None-unknown)</p>

### 2.9.2.2 Get Entrance Control Gate State by Gate Id [psgGate/gateStateById]

<p><b>Post Request URL</b></p>	<p>http://serverIP:serverPort/api/psgGate/gateStateById?gateId={gateId}&amp;access_token={apitoken}</p>
<p><b>Request Mode</b></p>	<p>GET</p>
<p><b>Request Content</b></p>	<p>None</p>
<p><b>Request Parameter Description</b></p>	<p>gateId: Entrance Control gate id, example: "8a888e8f820060eb018200e488650abb"</p>
<p><b>Response Result</b></p>	<p>{</p>

	<pre> "code": 0, "message": "success", "data": [   {     "id": "8a888e8f820060eb018200e488650abb",     "name": "10.8.14.234-1",     "deviceId": "8a888e8f820060eb018200e487a50a67",     "connect": "1",     "gateState": "0",     "alarm": "0",     "relay": "0"   } ] </pre>
<p><b>Response Result Description</b></p>	<p>Data: entrance control gate state array List;</p> <p>Id: gate id;</p> <p>Name: entrance control device name;</p> <p>DeviceId: device id;</p> <p>Connection: connection state, 0 offline, 1 online, 2 disabled;</p> <p>gateState: gate state (0 offline-device is not connect to entrance control gate, default: 0, 1 Online, 2 Disabled, 3 Lockdown)</p> <p>Alarm: alarm description (0-none, 1-be opened accidentally, 2-tampering, 3-duress password opening,4- duress fingerprint opening, 5-duress opening,6-Infrared ray exception, 7=gate error, None-unknown)</p> <p>relay: Relay state (0-off, 1-Exit open,2-Entry Open, None-unknown)</p>

### 2.9.2.3 Get Entrance Control Gate State by the SN of Device

[psgGate/gateStateBySn]

<p><b>Post Request URL</b></p>	<p>http://serverIP:serverPort/api/psgGate/gateStateBySn?deviceSn={sn}&amp;access_token={apitoken}</p>
<p><b>Request Mode</b></p>	<p>GET</p>

<b>Request Content</b>	None
<b>Request Parameter Description</b>	Sn: device sn
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": [     {       "id": "8a888e8f820060eb018200e488650abb",       "name": "10.8.14.234-1",       "deviceId": "8a888e8f820060eb018200e487a50a67",       "connect": "0",       "gateState": "0",       "alarm": "0",       "relay": "0"     }   ] } </pre>
<b>Response Result Description</b>	<p>Data: entrance control gate state array List;</p> <p>Id: gate id;</p> <p>Name: entrance control device name;</p> <p>DeviceId: device id;</p> <p>Connection: connection state, 0 offline, 1online, 2 disabled;</p> <p>gateState: gate state (0 offline-device is not connect to entrance control gate, default: 0, 1 Online, 2 Disabled, 3 Lockdown)</p> <p>Alarm: alarm description (0-none, 1-be opened accidentally, 2-tampering, 3-duress password opening,4- duress fingerprint opening, 5-duress opening,6-Infraed ray exception, 7=gate error, None-unknown)</p> <p>relay: Relay state (0-off, 1-Exit open,2-Entry Open, None-unknown)</p>

### 2.9.2.4 Get Entrance Control Gate Info by Gate Id [psgGate/getById]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgGate/getById?id={id}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	id: gate id;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "8a888e8f820060eb018200e488650abb",     "name": "10.8.14.234-1",     "deviceId": "8a888e8f820060eb018200e487a50a67"   } }</pre>
<b>Response Result Description</b>	data: gate info object; id: gate id; name: gate name; deviceId: entrance control device id;

### 2.9.2.5 Get Entrance Control Gate List [api/psgGate/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgGate/list?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	pageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0

<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "8a888e8f820060eb018200e488650abb",       "name": "10.8.14.234-1",       "deviceId": "8a888e8f820060eb018200e487a50a67"     }   ] }</pre>
<b>Response Result Description</b>	<p>data: entrance control gate array list;  id: gate id;  name: gate name;  deviceId: device id;</p>

### 2.9.2.6 Get Entrance Control Gate List (With Total Number) [api/v2/psgGate/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/psgGate/list?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: page, required, the value must be greater than 0;  pageSize, the number of each page, required, must be greater than 0</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 2,     "total": 1,   } }</pre>

	<pre> "data": [{   "id": "8a888e238488391e018488703f0a0066",   "name": "10.8.14.249-1",   "deviceId": "8a888e238488391e018488703e4b0012" }], "offset": 0, "lastPage": true } </pre>
<b>Response Result Description</b>	<p>data: the returned result;</p> <p>page: the current page number requested;</p> <p>size: the number of data requested by the current page number;</p> <p>total: the total number;</p> <p>data: entrance control gate array list;</p> <p>id: gate id;</p> <p>name: gate name;</p> <p>deviceId: device id;</p>

### 2.9.2.7 Remote Open by Gate Id [psgGate/remoteOpenById]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgGate/remoteOpenById?gateId={gateId}&openType={openType}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>gateId: Gate id, required; Example: "8a888e8f820060eb018200e488650abb"</p> <p>openType: Open gate type, 1-Open(Entry), 2-Open(Exit)</p>
<b>Response Result</b>	Refer to public response result, other can refer to Appendix-Error code
<b>Response Result Description</b>	public response parameter description

### 2.9.2.8 Remote Open by Gate Name [psgGate/remoteOpenByName]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgGate/remoteOpenByName?gateName={gateName}&openType={openType}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	gateName: entrance control gate name, required; openType: Open gate type, 1-Open(Entry), 2-Open(Exit)
<b>Response Result</b>	Refer to public response result, other can refer to Appendix-Error code
<b>Response Result Description</b>	public response parameter description

## 2.9.3 Entrance Control Gate Interface

### 2.9.3.1 Delete Person Level [psgLevel/deleteLevel]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgLevel/deleteLevel?pin={pin}&levelIds={levelIds}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Pin: pin, required Level Ids: level group id List, multiple comma separated, required, string; such as 402856aa6c3bbde1016c3bc2329200ff,402856aa6c3bbde1016c3bc232930100
<b>Response Result</b>	Refer to public Response Results
<b>Response Result Description</b>	Public response parameter description

### 2.9.3.2 Get Gate Control Level by Id [psgLevel/getById/{id}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgLevel/getById/{id}?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Id: level group id, required, string; such as "402856aa6c3bbde1016c3bc2329200ff"
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "402856aa6c3bbde1016c3bc2329200ff",     "name": "public level group"   } }</pre>
<b>Response Result Description</b>	Data: level group object; Id: level group id; Name: level group name;

### 2.9.3.3 Get Entrance Access Level by Name [psgLevel/getByName/{name}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgLevel/getByName/{name}?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Name: level group name, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {</pre>

	<pre>"id": "402856aa6c3bbde1016c3bc2329200ff", "name": "public level group" } }</pre>
<b>Response Result Description</b>	<p>Data: level group object;</p> <p>Id: level group id, string; such as '402856aa6c3bbde1016c3bc2329200ff'</p> <p>Name: level group name;</p>

#### 2.9.3.4 Get Gate Control Level by Id [psgLevel/getById/{id}](v2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/psgLevel/getById/{id}?access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Id: level group id, required, string; such as "402856aa6c3bbde1016c3bc2329200ff"
<b>Response Result</b>	<pre>{ "code": 0, "message": "success", "data": { "id": "402856aa6c3bbde1016c3bc2329200ff", "name": "public level group" } }</pre>
<b>Response Result Description</b>	<p>Data: level group object;</p> <p>Id: level group id;</p> <p>Name: level group name;</p>

#### 2.9.3.5 Get Entrance Access Level by Name [psgLevel/getByName/{name}](v2)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/psgLevel/getByName/{name}?access_token={apitoken}
-------------------------	-------------------------------------------------------------------------------------

<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Name: level group name, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "402856aa6c3bbde1016c3bc2329200ff",     "name": "public level group"   } }</pre>
<b>Response Result Description</b>	<p>Data: level group object;</p> <p>Id: level group id, string; such as '402856aa6c3bbde1016c3bc2329200ff'</p> <p>Name: level group name;</p>

### 2.9.3.6 Get Entrance Control Access Level [/psgLevel/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgLevel/list?pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {</pre>

	<pre>"id": "402856aa6c3bbde1016c3bc2329200ff", "name": "public level group" } ] }</pre>
<b>Response Result Description</b>	<p>Data: level group array list;</p> <p>Id: level group id, string; such as "402856aa6c3bbde1016c3bc2329200ff"</p> <p>Name: level group name;</p>

### 2.9.3.7 Sync Level Info to Device [psgLevel/syncLevel]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgLevel/syncLevel?levelId={levelId}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Level Id: level group id, required, string; such as "402856aa6c3bbde1016c3bc2329200ff"
<b>Response Result</b>	Refer to public response result, for any other result please refer to Appendix- <a href="#">Error code</a>
<b>Response Result Description</b>	Public response parameter description

### 2.9.3.8 Sync Person Level [/psgLevel/syncPerson]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgLevel/syncPerson?pin={pin}&levelIds={levelIds}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Level Ids: level group id list, multiple comma separated, required, string; such as "402856aa6c3bbde1016c3bc2329200ff"

	Pin: pin, required
<b>Response Result</b>	Refer to public response result, other can refer to Appendix-Error code
<b>Response Result Description</b>	Public response parameter description

### 2.9.3.9 Get Entrance Control Access Level (With Total Number)

[api/v2/psgLevel/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/psgLevel/list?pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	PageNo: page no, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 2,     "total": 1,     "data": [       {         "id": "402856aa6c3bbde1016c3bc2329200ff",         "name": "public level group"       }     ],     "offset": 0,     "lastPage": true   } }</pre>

	}
<b>Response Result Description</b>	<p>data: the returned result;</p> <p>page: the current page number requested;</p> <p>size: the number of data requested by the current page number;</p> <p>total: the total number;</p> <p>Data: level group array list;</p> <p>Id: level group id, string; such as "402856aa6c3bbde1016c3bc2329200ff"</p> <p>Name: level group name;</p>

### 2.9.3.10 Add Person Level By Level [api/v2/psgLevel/addLevelByLevel]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/psgLevel/addLevelByLevel?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>[   {     "levelId": "access level group id1",     "pins": [       "personnel ID pin1",       "personnel ID pin2",       "personnel ID pin3"     ]   },   {     "levelId": "access level group id2",     "pins": [       "personnel ID pin1",       "personnel ID pin2",       "personnel ID pin3"     ]   } ]</pre>

<b>Request Parameter Description</b>	levelId: access level group id, required; Pins: personnel ID pin list, required;
<b>Response Result</b>	Refer to public Response Results
<b>Response Result Description</b>	Public response parameter description

### 2.9.3.11 Add Person Level By Person [api/v2/psgLevel/addLevelByPerson]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/psgLevel/addLevelByPerson?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>[   {     "pin": "personnel ID pin1",     "levelIds": [       "access level group id1",       "access level group id2",       "access level group id3"     ]   },   {     "pin": "personnel ID pin2",     "levelIds": [       "access level group id1",       "access level group id2",       "access level group id3"     ]   } ]</pre>
<b>Request Parameter Description</b>	Pin: personnel ID pin list, required; levelIds: access level group id, required;
<b>Response Result</b>	Refer to public Response Results

<b>Response Result Description</b>	Public response parameter description
------------------------------------	---------------------------------------

## 2.9.4 Entrance Control Gate Reader Interface

### 2.9.4.1 Get Entrance Control Gate Reader Info List[psgReader/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgReader/list?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	PageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "8a888e8f820060eb018200e488660abc",       "gateId": "8a888e8f820060eb018200e488650abb",       "name": "10.8.14.234-1-入",       "readerNo": 1,       "readerState": 0     },     {       "id": "8a888e8f820060eb018200e488660abd",       "gateId": "8a888e8f820060eb018200e488650abb",       "name": "10.8.14.234-1-出",       "readerNo": 2,       "readerState": 1     }   ] }</pre>

	<pre> ] } </pre>
<b>Response Result Description</b>	<p>Data: reader array List;</p> <p>Id: reader id;</p> <p>gateld: entrance control gate id;</p> <p>Name: reader name;</p> <p>ReaderNo: readerNo;</p> <p>ReaderState: reader state, 0: in, 1: out;</p>

### 2.9.4.2 Get Entrance Control Gate Reader Info List (With Total Number)

[api/v2/psgReader/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/psgReader/list?pageNo=1&pageSi ze=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: page, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0</p>
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 2,     "total": 2,     "data": [{       "id": "8a888e238488391e018488703f0b0067",       "gateld": "8a888e238488391e018488703f0a0066",       "name": "10.8.14.249-1-入",       "readerNo": 1,       "readerState": 0     }   ] } </pre>

	<pre> }, {   "id": "8a888e238488391e018488703f0b0068",   "gateId": "8a888e238488391e018488703f0a0066",   "name": "10.8.14.249-1-出",   "readerNo": 2,   "readerState": 1 } ], "offset": 0, "lastPage": true } } </pre>
<p><b>Response Result Description</b></p>	<p>data: the returned result;  page: the current page number requested;  size: the number of data requested by the current page number;  total: the total number;  Data: reader array List;  Id: reader id;  gateId: entrance control gate id;  Name: reader name;  ReaderNo: readerNo;  ReaderState: reader state, 0: in, 1: out;</p>

### 2.9.4.3 Get Entrance Control Gate Reader Info by Reader Id [psgReader/getById]

<p><b>Post Request URL</b></p>	<p>http://serverIP:serverPort/api/psgReader/getById?id={id}&amp;access_to ken={apitoken}</p>
<p><b>Request Mode</b></p>	<p>GET</p>
<p><b>Request Content</b></p>	<p>None</p>
<p><b>Request Parameter Description</b></p>	<p>id: reader id, required;</p>

<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "402856aa6c3c5063016c3cb3a0360005",     "gateId": "402856aa6bff4b1a016bff5374800c0e",     "name": "192.168.214.15-1-in",     "readerNo": 1,     "readerState": 0   } }</pre>
<b>Response Result Description</b>	<p>data: reader info object;</p> <p>id: reader id;</p> <p>Gate Id: gate id;</p> <p>name: reader name;</p> <p>readerNo: reader no;</p> <p>readState: reader state, 0: in, 1:out;</p>

## 2.9.5 Entrance Control Device transaction Interface

### 2.9.5.1 Get Entrance Control Transaction by Device Sn

[api/psgTransaction/device/{deviceSn}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgTransaction/device/{deviceSn}?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: pageNo, required, the value must be greater than 0;</p> <p>PageSize: required, the value must be greater than 0, not more than 1000 per page;</p> <p>DeviceSn: device sn;</p>

	<p>startDate: stat time, not required; endDate: end time, not required;</p>
<p><b>Response Result</b></p>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "8a888e8f821044f3018210ce84150029",       "eventTime": "2022-07-18 18:15:42",       "pin": "",       "name": "",       "lastName": "",       "deptName": "",       "areaName": "Area Name",       "cardNo": "",       "devSn": "CKN4213360081",       "verifyModeName": "Others",       "eventName": "The entrance control gate is not connected",       "eventPointName": "10.8.14.234-1",       "readerName": "Others",       "devName": "10.8.14.234",       "logId": 3603,       "eventNo": 110,       "gateType": 0     }   ] }</pre>
<p><b>Response Result Description</b></p>	<p>data: transaction array list; id: event id, "402856aa6c032ca3016c03351d2600fd"; eventTime: record device trigger time; pin: pin; name: person name;</p>

	<p>lastName: person last name, (English);</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p> <p>eventNo: Event Number;</p> <p>gateType: Gate Type, 0-Unknown(Device is not connected to Gate);1-Tripod Turnstile; 2-Flap Barrier;3-Swing Barrier</p>
--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.9.5.2 Get Entrance Control Transaction by Device Sn (With Total Number)

[api/v2/psgTransaction/device/{deviceSn}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/psgTransaction/device/{deviceSn}?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: pageNo, required, the value must be greater than 0;</p> <p>PageSize: required, the value must be greater than 0, not more than 1000 per page;</p> <p>DeviceSn: device sn;</p> <p>startDate: stat time, not required;</p> <p>endDate: end time, not required;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success", }</pre>

```
"data": {  
  
  "page": 1,  
  
  "size": 2,  
  
  "total": 1,  
  
  "data": [{  
  
    "id": "8a888e8f821044f3018210ce84150029",  
  
    "eventTime": "2022-07-18 18:15:42",  
  
    "pin": "",  
  
    "name": "",  
  
    "lastName": "",  
  
    "deptName": "",  
  
    "areaName": "Area Name",  
  
    "cardNo": "",  
  
    "devSn": "CKN4213360081",  
  
    "verifyModeName": "Others",  
  
    "eventName": "The entrance control gate is not  
connected",  
  
    "eventPointName": "10.8.14.234-1",  
  
    "readerName": "Others",  
  
    "devName": "10.8.14.234",  
  
    "logId": 3603,  
  
    "eventNo": 110,  
  
    "gateType": 0
```

	<pre>     }},     "offset": 0,     "lastPage": true   } } </pre>
<p><b>Response Result Description</b></p>	<p>data: the returned result;</p> <p>page: the current page number requested;</p> <p>size: the number of data requested by the current page number;</p> <p>total: the total number;</p> <p>data: transaction array list;</p> <p>id: event id, "402856aa6c032ca3016c03351d2600fd";</p> <p>eventTime: record device trigger time;</p> <p>pin: pin;</p> <p>name: person name;</p> <p>lastName: person last name, (English);</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p> <p>eventNo: Event Number;</p> <p>gateType: Gate Type, 0-Unknown(Device is not connected to Gate);1-Tripod Turnstile; 2-Flap Barrier;3-Swing Barrier</p>

### 2.9.5.3 Get Entrance Control Transaction by Device Sn (With Total Number-v3)

[api/v3/psgTransaction/device]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v3/psgTransaction/device/{deviceSn}?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: pageNo, required, the value must be greater than 0;</p> <p>PageSize: required, the value must be greater than 0, not more than 1000 per page;</p> <p>DeviceSn: device sn;</p> <p>startDate: stat time, not required;</p> <p>endDate: end time, not required;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 1,     "size": 2,     "total": 1,     "data": [{       "id": "8a888e8f821044f3018210ce84150029",       "eventTime": "2022-07-18 18:15:42",       "pin": "",       "name": "",       "lastName": ""     }   ] }</pre>

	<pre> "deptName": "", "areaName": "Area Name", "cardNo": "", "devSn": "CKN4213360081", "verifyModeName": "Others", "eventName": "The entrance control gate is not connected", "eventPointName": "10.8.14.234-1", "readerName": "Others", "devName": "10.8.14.234", "logId": 3603, "eventNo": 110, "gateType": 0 }}, "offset": 0, "lastPage": true } } </pre>
<p><b>Response Result Description</b></p>	<p>data: the returned result;  page: the current page number requested;  size: the number of data requested by the current page number;  total: the total number;  data: transaction array list;  id: event id, "402856aa6c032ca3016c03351d2600fd";  eventTime: record device trigger time;  pin: pin;</p>

	<p>name: person name;</p> <p>lastName: person last name, (English);</p> <p>deptName: department name;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p> <p>eventNo: Event Number;</p> <p>gateType: Gate Type, 0-Unknown(Device is not connected to Gate);1-Tripod Turnstile; 2-Flap Barrier;3-Swing Barrier</p>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

#### 2.9.5.4 Get Entrance Control Gate Transaction List[api/psgTransaction/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgTransaction/list?personPin={personPin}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>personPin: personPin, [not required]</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",</pre>

	<pre> "data": [   {     "id": "8a888e8f821044f3018210ce84150029",     "eventTime": "2022-07-18 18:15:42",     "pin": "",     "name": "",     "lastName": "",     "deptName": "",     "areaName": "Area Name",     "cardNo": "",     "devSn": "CKN4213360081",     "verifyModeName": "Other",     "eventName": "The Gate is not connected",     "eventPointName": "10.8.14.234-1",     "readerName": "Others",     "devName": "10.8.14.234",     "logId": 3603,     "eventNo": 110,     "gateType": 0   } ] </pre>
<p><b>Response Result Description</b></p>	<p>data: transaction array list;  id: event ID;  eventTime: record device trigger time;  pin: pin;  name: person name;  lastName: last name(English);  deptName: department name;  areaName: area name;  cardNo: card no;  devSn: device sn;  verifyModeName: verification mode name;</p>

	<p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transactionindex value, there will be a value if device support it;</p> <p>eventNo: Event Number;</p> <p>gateType: Gate Type, 0-Unknown(Device is not connected to Gate);1-Tripod Turnstile; 2-Flap Barrier;3-Swing Barrier</p>
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.9.5.5 Get Entrance Control Gate Transaction (With Total Number)List[api/v2/psgTransaction/list]

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/psgTransaction/list?personPin={personPin}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>personPin:personPin, [not required]</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 1,     "size": 1,     "total": 1,     "data": [{       "id": "8a888e8f821044f3018210ce84150029",</pre>

	<pre> "eventTime": "2022-07-18 18:15:42", "pin": "", "name": "", "lastName": "", "deptName": "", "areaName": "Area Name", "cardNo": "", "devSn": "CKN4213360081", "verifyModeName": "Other", "eventName": "The Gate is not connected", "eventPointName": "10.8.14.234-1", "readerName": "Other", "devName": "10.8.14.234", "logId": 3603, "eventNo": 110, "gateType": 0 }}, "offset": 0, "lastPage": true } } </pre>
<p><b>Response Result Description</b></p>	<p>data: the returned result;  page: the current page number requested;  size: the number of data requested by the current page number;  total: the total number;  data: transaction array list;  id: event ID;  eventTime: record device trigger time;  pin: pin;  name: person name;  lastName: last name(English);  deptName: department name;  areaName: area name;</p>

	<p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transactionindex value, there will be a value if device support it;</p> <p>eventNo: Event Number;</p> <p>gateType: Gate Type, 0-Unknown(Device is not connected to Gate);1-Tripod Turnstile; 2-Flap Barrier;3-Swing Barrier</p>
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.9.5.6 Get Entrance Control Device Real-Time Monitoring Transaction

[api/psgTransaction/monitor]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgTransaction/monitor?timestamp={timestamp}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Timestamp: timestamp millisecond, such as 1494467276539, required; Get frequency cannot be too high, it is recommended to obtain at least 10 seconds or more; there is no need to change after timestamp value set the first request, it is allowed to keep using the first request
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": null,       "eventTime": "2022-07-18 18:13:33", </pre>

	<pre> "pin": "", "name": "", "lastName": "", "deptName": "", "areaName": "Area Name", "cardNo": "", "devSn": "CKN4213360081", "verifyModeName": "Other", "eventName": "Gate is not connected", "eventPointName": "10.8.14.234-1", "readerName": "Other", "devName": "10.8.14.234", "logId": 3601, "eventNo": 110, "gateType": 0     }   ] } </pre>
<p><b>Response Result Description</b></p>	<p>data:transaction monitoring record array List;  Id: event id(Not stored in database yet, so it is null)  eventTime: record device trigger time;  pin:pin;  areaName: area name;  cardNo: card no;  devSn: device sn;  verifyModeName: verification mode name;  eventName: transaction description;  eventPointName: transaction point name;  readerName: reader name;  devName: device name;  logId: transactionindex value, there will be a value if device support it;  eventNo: Event Number;  gateType: Gate Type, 0-Unknown(Device is not connected to</p>

Gate);1-Tripod Turnstile; 2-Flap Barrier;3-Swing Barrier

### 2.9.5.7 Get Personnel Entrance Control Access Transaction by Pin

[api/psgTransaction/person/{pin}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/psgTransaction/person/{pin}?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p> <p>pin: pin, required;</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "8a888e8f821044f3018210ce84150029",       "eventTime": "2022-07-18 18:15:42",       "pin": "",       "name": "",       "lastName": "",       "deptName": "",       "areaName": "Area Name",       "cardNo": "",       "devSn": "CKN4213360081",       "verifyModeName": "Other",       "eventName": "The Gate is not connected",       "eventPointName": "10.8.14.234-1",     }   ] }</pre>

	<pre> "readerName": "Others", "devName": "10.8.14.234", "logId": 3603, "eventNo": 110, "gateType": 0     }   ] } </pre>
<p><b>Response Result Description</b></p>	<p>data:transaction monitoring record array List;  Id: event id(Not stored in database yet, so it is null)  eventTime: record device trigger time;  pin: pin;  areaName: area name;  cardNo: card no;  devSn: device sn;  verifyModeName: verification mode name;  eventName: transaction description;  eventPointName: transaction point name;  readerName: reader name;  devName: device name;  logId: transactionindex value, there will be a value if device support it;  eventNo: Event Number;  gateType: Gate Type, 0-Unknown(Device is not connected to Gate);1-Tripod Turnstile; 2-Flap Barrier;3-Swing Barrier</p>

### 2.9.5.8 Get Entrance Control Gate Transaction List (With Total Number)

[api/v2/psgTransaction/list]

<p><b>Post Request URL</b></p>	<p>http://serverIP:serverPort/api/v2/psgTransaction/list?personPin={personPin}&amp;startDate={startDate}&amp;endDate={endDate}&amp;pageNo=1&amp;pageSize=20&amp;access_token={apitoken}</p>
<p><b>Request Mode</b></p>	<p>GET</p>

<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>personPin: personPin, [not required]</p> <p>startDate: startDate, [not required]</p> <p>endDate: endDate, [not required]</p> <p>PageNo: page no, required, the value must be greater than 0;</p> <p>pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 1,     "size": 1,     "total": 1,     "data": [{       "id": "8a888e8f821044f3018210ce84150029",       "eventTime": "2022-07-18 18:15:42",       "pin": "",       "name": "",       "lastName": "",       "deptName": "",       "areaName": "Area",       "cardNo": "",       "devSn": "CKN4213360081",       "verifyModeName": "Others",       "eventName": "The gate is not connected",       "eventPointName": "10.8.14.234-1",       "readerName": "Other",       "devName": "10.8.14.234",       "logId": 3603,       "eventNo": 110,       "gateType": 0     }],   } }</pre>

	<pre> "offset": 0, "lastPage": true } } </pre>
<p><b>Response Result Description</b></p>	<p>data: the returned result;  page: the current page number requested;  size: the number of data requested by the current page number;  total: the total number;  data: transaction array list;  id: 5;  eventTime: record device trigger time;  pin: pin;  name: person name;  lastName: last name;  deptName: department name;  areaName: area name;  cardNo: card no;  devSn: device sn;  verifyModeName: verification mode name;  eventName: transaction description;  eventPointName: transaction point name;  readerName: reader name;  devName: device name;  logId: transaction index value, there will be a value if device support it;  eventNo: Event Number;  gateType: Gate Type, 0-Unknown(Device is not connected to Gate);1-Tripod Turnstile; 2-Flap Barrier;3-Swing Barrier</p>

### 2.9.5.9 Get Personnel Entrance Control Access Transaction by Pin (With Total Number) [api/v3/psgTransaction/person/{pin}]

<p><b>Post Request URL</b></p>	<p>http://serverIP:serverPort/api/v3/psgTransaction/person/{pin}?pageNo=1&amp;pageSize=20&amp;access_token={apitoken}</p>
--------------------------------	---------------------------------------------------------------------------------------------------------------------------

<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>PageNo: page no, required, the value must be greater than 0;          pageSize, the number of each page, required, must be greater than 0, not more than 1000 per page;          pin: pin, required;          startDate: startDate, [not required]          endDate: endDate, [not required]</p>
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 1,     "size": 1,     "total": 1,     "data": [{       "id": "8a888e8f821044f3018210959f620007",       "eventTime": "2022-07-18 17:13:36",       "pin": "1",       "name": "",       "lastName": "",       "deptName": "",       "areaName": "Area Name",       "cardNo": "",       "devSn": "CKN4213360081",       "verifyModeName": "Others",       "eventName": "Connection abort",       "eventPointName": "",       "readerName": "Other",       "devName": "10.8.14.234",       "logId": -1,       "eventNo": 700,       "gateType": 0     }   ] }</pre>

	<pre>         }],         "offset": 0,         "lastPage": true       }     }   } </pre>
<p><b>Response Result Description</b></p>	<p>data: the returned result;</p> <p>page: the current page number requested;</p> <p>size: the number of data requested by the current page number;</p> <p>total: the total number;</p> <p>data:transaction monitoring record array List;</p> <p>Id: event id (Not stored in database yet, so it is null)</p> <p>eventTime: record device trigger time;</p> <p>pin: pin;</p> <p>areaName: area name;</p> <p>cardNo: card no;</p> <p>devSn: device sn;</p> <p>verifyModeName: verification mode name;</p> <p>eventName: transaction description;</p> <p>eventPointName: transaction point name;</p> <p>readerName: reader name;</p> <p>devName: device name;</p> <p>logId: transaction index value, there will be a value if device support it;</p> <p>eventNo: Event Number;</p> <p>gateType: Gate Type, 0-Unknown(Device is not connected to Gate);1-Tripod Turnstile; 2-Flap Barrier;3-Swing Barrier</p>

## 2.10 Facekiosk

### 2.10.1 FaceKiosk Personal Information

#### 2.10.1.1 Edit Personal Information

<b>Post Request URL</b>	http://serverIP:serverPort /api/attPerson/editInsPerson?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	{ "category": 0, "perDevAuth": 0, "personPin": 1 }
<b>Request Parameter Description</b>	category (string): ID category (0=Ordinary staff, 1=VIP, Blacklist) perDevAuth (integer): Device operation authority (0=Ordinary staff,14=super administrators) personPin (string): Personal number
<b>Response Result</b>	{ "code": 0, "message": "success", "data": null }

### 2.10.2 FaceKiosk Area Personal

#### 2.10.2.1 Delete Area Personnel

<b>Post Request URL</b>	http://serverIP:serverPort/api/insAreaPerson/delAreaPerson?access_token={apitoken}
-------------------------	------------------------------------------------------------------------------------

<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "code": 1,   "pins": "1,2,3" }</pre>
<b>Request Parameter Description</b>	<p>code(string): Area code (For details/api/insAreaPerson/getAreaList)</p> <p>pins (string): Personnel number collection(Multiple personnel are separated by ',')</p>
<b>Response Result</b>	<pre>{   "code":0,   "message":"success",   "data": null }</pre>
<b>Response Result Description</b>	

### 2.10.2.2 Add Area Personnel

<b>Post Request URL</b>	http://serverIP:serverPort/api/insAreaPerson/setAreaPerson?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	<pre>{   "code": 1,   "pins": "1,2,3" }</pre>

	}
<b>Request Parameter Description</b>	<p>code (string): Area code (For details/api/insAreaPerson/getAreaList)</p> <p>pins (string): personnel number collection(Multiple personnel are separated by ',')</p>
<b>Response Result</b>	<pre>{   "code":0,   "message":"success",   "data": null }</pre>
<b>Response Result Description</b>	

### 2.10.2.3 Get a List of Attendance Zones

<b>Post Request URL</b>	http://serverIP:serverPort/api/insAreaPerson/getAreaList?pageNo=1&pageSize=10&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	
<b>Request Parameter Description</b>	
<b>Response Result</b>	<pre>{   "code":0,   "message":"success",   "data": [</pre>

	<pre>{ "id": "402855b27d4b9eb4017d4ba051690002",   "code": "1",   "name": "Area name" }] }</pre>
<b>Response Result Description</b>	<p>id: Area ID</p> <p>code: Area code</p> <p>name: Area name</p>

#### 2.10.2.4 Get a list of attendance zones (With total number)

<b>Post Request URL</b>	http://serverIP:serverPort/api/v2/insAreaPerson/getAreaList?pageNo=1&pageSize=10&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	
<b>Request Parameter Description</b>	
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 1,     "total": 2,     "data": [{</pre>

	<pre>         "id": "402855b27d4b9eb4017d4ba051690002",         "code": "1",         "name": "Area name"       }     ],     "offset": 0,     "lastPage": false   } } </pre>
<b>Response Result Description</b>	<p>id: Area ID</p> <p>code: Area code</p> <p>name: Area name</p>

## 2.10.3 FaceKiosk Device Interface

### 2.10.3.1 Get Facekiosk Device List [api/device/attAdDeviceList]

<b>Post Request URL</b>	http://serverIP:serverPort/api/device/attAdDeviceList?pageNo=1&pageSize=20&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	<p>pageNo: pageno, required, the value must be greater than 0;</p> <p>pageSize: the number of each page, required, must be greater than 0</p>
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": [ </pre>

	<pre>{   "id": "402848327562dedd01756d555c726227",   "sn": "6271203100008",   "name": "FaceKiosk-1",   "type": "Robot-Pad",   "status": "1",   "module": "ins" } {   "id": "4028483274b4eeb40174d3e117bf4329",   "sn": "174559110004",   "name": "174559110004",   "type": "ZK-S1050",   "status": "1",   "module": "ins" } ] }</pre>
<p><b>Response Result Description</b></p>	<p>Data: device array list;          Id: device id;          Sn: device sn;          Name: device name;          state: device state, 1 enabled, 0 disabled;          Module: module, acc- Access, att-Attendance, ele- Elevator, ins-FaceKiosk;</p>

### 2.10.3.2 Get Facekiosk Device List (With Total Number)

[api/device/attAdDeviceList]

<p><b>Post Request URL</b></p>	<p>http://serverIP:serverPort/api/v2/device/attAdDeviceList?pageNo=1&amp;pageSize=20&amp;access_token={apitoken}</p>
<p><b>Request Mode</b></p>	<p>GET</p>
<p><b>Request Content</b></p>	<p>None</p>

<b>Request Parameter Description</b>	<p>pageNo: pageno, required, the value must be greater than0;          pageSize: the number of each page, required, must be greater than 0</p>
<b>Response Result</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 1,     "total": 2,     "data": [{       "id": "402848327562dedd01756d555c726227",       "sn": "6271203100008",       "name": "FaceKiosk-1",       "type": "Robot-Pad",       "status": "1",       "module": "ins"     }   ],   "offset": 0,   "lastPage": false } </pre>
<b>Response Result Description</b>	<p>Data: device array list;          Id: device id;          Sn: device sn;</p>

	Name: device name; state: device state, 1 enabled, 0 disabled; Module: module, acc- Access, att-Attendance, ele- Elevator, ins-FaceKiosk;
--	-------------------------------------------------------------------------------------------------------------------------------------------------

## 2.10.4 Media Interface

### 2.10.4.1 Add or Update Media Info[adMedia/add]

<b>Post Request URL</b>	http://serverIP:serverPort/api/adMedia/add&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request Content</b>	File: "bayue.png", mediaName: "Media name" (be included in a form data to submit)
<b>Request Parameter Description</b>	File: Media file, required mediaName: media name, required (be included in form data to send)
<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.10.4.2 Delete Media by Name[adMedia/delete/{name}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/adMedia/delete/{name}&access_token={apitoken}
<b>Request Mode</b>	DELETE
<b>Request Content</b>	None
<b>Request Parameter Description</b>	Name: media name, required;

<b>Response Result</b>	Refer to the General Response Result <a href="#">Interface Parameter</a> . For more details, see Appendix - <a href="#">Error Codes</a>
<b>Response Result Description</b>	Refer to General Response Result Parameter Description <a href="#">Interface Parameter</a>

### 2.10.4.3 Get Media Object by Name[adMedia/get/{name}]

<b>Post Request URL</b>	http://serverIP:serverPort/api/adMedia/get/{name}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request Content</b>	None
<b>Request Parameter Description</b>	name: media name, required;
<b>Response Result</b>	<pre>{   "code": 0,   "message": "success",   "data": [     {       "id": "402856aa6c3bbde1016c3bc2329200ff",       "mediaName": "111",       "mediaSize": "21KB",       "mediaSuffix": "png",       "mediaType": "1",       "mediaPath": "/att/picture/1535520932078.png"     }   ] }</pre>
<b>Response Result Description</b>	Data: media object Id: media id; such as: "402856aa6c3bbde1016c3bc2329200ff" mediaName: media name; mediaSize: media size;

mediaType: media type; 1: picture; 2: video  
 mediaPath: media path

## 2.11 Smart Video Surveillance

### 2.11.1 Smart Video Device Interface

#### 2.11.1.1 Obtain the Main Device Interface

<b>Post Request URL</b>	http://serverIP:serverPort/api/ivsDevice/getParentDevList?pageNo=1&pageSize=20 &access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	not have
<b>Request Parameter Description</b>	pageNo: Page number, required, must be greater than 0; pageSize: Number of articles on each page, required, must be greater than 0;
<b>Response Results</b>	{ "code": 0, "message": "success", "data": { "page": 0, "size": 20, "total": 1, "data": [ {

	<pre> "id": "8a888ee28dba4ebc018dba51d1180001",  "ip": "10.8.14.126",  "status": "1",  "userName": "admin",  "passWord": "eav1a\$calRqfngfSSP5RIQqTjlsA==",  "port": "8081",  "type": "7"  }  ],  "offset": 0,  "lastPage": true  }  } </pre>
<p><b>Response Result Description</b></p>	<p>Id: primary device id</p> <p>Ip: primary device ip</p> <p>status: Online status</p> <p>userName: Account name</p> <p>passWord: password</p> <p>port: port</p> <p>Type: Primary device type</p>

### 2.11.1.2 Get the Device Interface

<b>Post Request</b>	http://serverIP:serverPort/api/ivsDevice/getDevList?pageNo=1&pageSize=20&paren
---------------------	--------------------------------------------------------------------------------

<b>URL</b>	tId=[parentId]&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	not have
<b>Request Parameter Description</b>	<p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize: Number of articles on each page, required, must be greater than 0;</p> <p>parentId: Main device id, required</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 1,     "total": 18,     "data": [       {         "id": "8a888ee28dba4ebc018dba51d67a0003",         "cameraName": "CH1",         "cameraType": "DOME",         "status": "0",         "channel": "1",         "parentType": "7",         "channelCode": "CH1", </pre>

	<pre> "vidParentDeviceId": "8a888ee28dba4ebc018dba51d1180001", "channelId": "8a888ee28dba4ebc018dba524f6f0015" } ], "offset": 0, "lastPage": false } } </pre>
<b>Response Result Description</b>	<p>Id: equipment id</p> <p>cameraName: Device name</p> <p>cameraType: Device type</p> <p>status: Online status</p> <p>channel: Channel number</p> <p>parentType: Main device type</p> <p>channelCode: Channel number</p> <p>vidParentDeviceId: Main device id</p> <p>channelId: channel id</p>

### 2.11.1.3 Gets the Device Live Stream Address

<b>Post Request URL</b>	<p>http://serverIP:serverPort/api/ivsDevice/getDeviceRtsp?channelId=[channelId]&amp;rtspType=[rtspType]&amp;access_token={apitoken}</p>
<b>Request Mode</b>	<p>GET</p>
<b>Request content</b>	<p>not have</p>

<b>Request</b> <b>Parameter</b> <b>Description</b>	channelId: Channel id, required; rtspType: Code stream type, required, 0: all code stream, 1: main code stream, 2: auxiliary code stream;
<b>Response</b> <b>Results</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "resultCode": "0",     "content": {       "rtspUrls": [         {           "rtspUrl": "rtsp://admin:Admin123@10.8.14.126:554/ch02.264?type=udp",           "type": 1,           "parentType": 7         },         {           "rtspUrl": "rtsp://admin:Admin123@10.8.14.126:554/ch02_sub.264?type=udp",           "type": 2,           "parentType": 7         }       ]     }   } } </pre>

	} }
<b>Response Result Description</b>	rtspUrls: Code stream list rtspUrl: Code stream address Type: code stream type parentType: Main device type

#### 2.11.1.4 Gets the Device Playback List

<b>Post Request URL</b>	http://serverIP:serverPort/api/ivsDevice/getVideoList?channelId=[channelId]&startTime=[startTime]&endTime=[endTime]&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	not have
<b>Request Parameter Description</b>	channelId: Channel id, required; startTime: Start time, required, example: 2024-02-19 00:00:00; endTime: End time, required, example: 2024-02-19 16:37:40;
<b>Response Results</b>	{ "code": 0, "message": "success", "data": { "videoItem": [ { "startTime": "2024-02-19 05:41:02", "endTime": "2024-02-19 07:44:32", "type": "PLAN" }

	<pre> }, {   "startTime": "2024-02-19 07:45:40",   "endTime": "2024-02-19 16:37:40",   "type": "PLAN" } ] } } </pre>
<b>Response Result Description</b>	<p>videoItemList: Playback list</p> <p>startTime: Start time</p> <p>endTime: End time</p> <p>Type: video type, PLAN: planned video, ALARM: alarm video</p>

### 2.11.1.5 Obtain the Device Back to the Release Address

<b>Post Request URL</b>	<p>http://serverIP:serverPort/api/ivsDevice/getRtspPlayBack?channelId=[channelId]&amp;startTime=[startTime]&amp;endTime=[endTime]&amp;access_token={apitoken}</p>
<b>Request Mode</b>	<p>GET</p>
<b>Request content</b>	<p>not have</p>
<b>Request Parameter Description</b>	<p>channelId: Channel id, required;</p> <p>startTime: Start time, required, example: 2024-02-19 00:00:00;</p> <p>endTime: End time, required, example: 2024-02-19 16:37:40; (the time interval is in</p>

	the list obtained in the previous interface, no video can not be broadcast)
<b>Response Results</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "passWord": "Admin123",     "rtspUrl":       "rtsp://admin:Admin123@10.8.14.126:554/recording?ch=2&amp;stream=1&amp;start=20240219074432&amp;stop=20240219054102",     "vidParentDeviceId": "8a888ee28dba4ebc018dba51d1180001",     "rtspUrls": [       {         "rtspUrl":           "rtsp://admin:Admin123@10.8.14.126:554/recording?ch=2&amp;stream=0&amp;start=20240219074432&amp;stop=20240219054102",         "type": 1       },       {         "rtspUrl":           "rtsp://admin:Admin123@10.8.14.126:554/recording?ch=2&amp;stream=1&amp;start=20240219074432&amp;stop=20240219054102",         "type": 2       }     ],     "type": "7", </pre>

	<pre>"userName": "admin", "channelCode": "CH2" } }</pre>
<b>Response Result Description</b>	<p>rtspUrls: Code stream list</p> <p>rtspUrl: Code stream address</p> <p>Type: code stream type</p> <p>vidParentDeviceId: Main device id</p> <p>userName: Account name</p> <p>passWord: password</p> <p>channelCode: Channel number</p> <p>Type: Primary device type</p>

### 2.11.1.6 Set the Push Address

<b>Post Request URL</b>	http://serverIP:serverPort/api/ivsDevice/setPushSetting?pushAddress={pushAddress}&alarmCodes={alarmCodes}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	not have
<b>Request Parameter Description</b>	<p>pushAddress: Push address, required;</p> <p>alarmCodes: Alarm code, select to fill in, comma separated, do not fill in, subscribe to all types of alarm;</p>
<b>Response Results</b>	<pre>{ "code": 0,</pre>

	<pre>"message": "success", "data": null }</pre>
<b>Response Result Description</b>	

## 2.12 Intrusion Alarm

### 2.12.1 Intrusion Event Record Interface

#### 2.12.1.1 Intrusion Event Record Interface

<b>Post Request URL</b>	<pre>http://serverIP:serverPort/api/ias/list?access_token={apitoken}&amp;deviceName={deviceName}&amp;partitionName={partitionName}&amp;pointName={pointName}&amp;startTime={startTime}&amp;endTime={endTime}&amp;pageNo={pageNo}&amp;pageSize={pageSize}</pre>
<b>Request Mode</b>	GET
<b>Request content</b>	None
<b>Request Parameter Description</b>	<pre>deviceName:Device Name; partitionName:Partition Name; pointName:Zone Name; startTime:Start time of time range query; endTime:End time of time range query; pageNo:Page number, required, must be greater than 0; pageSize:Number of items per page, required, must be greater than 0, and must not</pre>

	exceed 1000 items per page;
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 1,     "size": 2,     "total": 541,     "data": [       {         "sortName": null,         "sortOrder": null,         "equals": false,         "id": "8a8890109909f8df019909feaaad002d",         "deviceName": "horn",         "deviceIp": "10.8.16.156",         "eventTime": "2025-09-02 18:35:13",         "eventName": "Disarm",         "areaName": "Area Name",         "partitionName": "1",         "pointName": "",         "eventContent": "2025-09-02 18:35:13 Partition 1 Disarm",</pre>

```
"eventCid": "1401",  
  
"eventTypeName": "Disarm Event"  
  
},  
  
{  
  
"sortName": null,  
  
"sortOrder": null,  
  
"equals": false,  
  
"id": "8a8890109909f8df019909feaa9b002b",  
  
"deviceName": "horn",  
  
"deviceIp": "10.8.16.156",  
  
"eventTime": "2025-09-02 18:35:13",  
  
"eventName": "Alarm Cancel",  
  
"areaName": "Area Name",  
  
"partitionName": "1",  
  
"pointName": "",  
  
"eventContent": "2025-09-02 18:35:13 Partition 1 Alarm Cancel",  
  
"eventCid": "1406",  
  
"eventTypeName": "Disarm Event"  
  
}  
  
],  
  
"offset": 2,  
  
"lastPage": false  
  
}
```

	<pre> } </pre>
<b>Response Result Description</b>	<pre> data:array sets;  id:event id;  deviceName:Device Name;  deviceIp: Device IP address;  eventTime:Event time;  eventName:Event Name;  areaName:area name;  partitionName:Partition Name;  pointName:Zone Name;  eventContent:Event content;  eventCid:Event CID;  eventTypeName:Event type name; </pre>

## 2.13 Space Management

### 2.13.1 Space Device Interface

#### 2.13.1.1 Get Space Device Info by Sn

<b>Post Request URL</b>	<pre> http://serverIP:serverPort/api/spaceDevice/getBySn?sn={sn}&amp;access_token={apitoken} </pre>
<b>Request Mode</b>	GET
<b>Request content</b>	None

<b>Request Parameter Description</b>	Sn: device serial number, required;
<b>Response Results</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "id": "8a8890e192d0aeec0192dc804b356b92",     "sn": "500152130033",     "name": "10.8.16.251",     "type": "FaceKiosk-H10A",     "status": "0",     "module": "space"   } } </pre>
<b>Response Result Description</b>	Data: device information; Id: device id; Sn: device sn; Name: device name; Type: device model; Status: device status, 0: offline, 1: online, 2: disabled; Module: the belonging module, space: space management module;

### 2.13.1.2 Get Device List

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceDevice/list?access_token={apitoken}&deviceName={deviceName}&sn={sn}&pageNo={pageNo}&pageSize={pageSize}
<b>Request Mode</b>	GET
<b>Request content</b>	None
<b>Request Parameter Description</b>	<p>deviceName: device name;</p> <p>Sn: device serial number, required;</p> <p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize: The number of each page, required, must be greater than 0, each page shall not exceed 1000;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 50,     "total": 1,     "data": [       {         "id": "8a8890e192e039c60192e073913f4675",         "sn": "500152110018",         "name": "10.8.53.104", </pre>

	<pre>                 "type": "FaceKiosk-H10A",                 "status": "0",                 "module": "space"             }         ],         "offset": 0,         "lastPage": true     } } </pre>
<p><b>Response Result Description</b></p>	<p>Data: array sets;</p> <p>Id: device id;</p> <p>Sn: device sn;</p> <p>Name: device name;</p> <p>Type: device model;</p> <p>Status: device status, 0: offline, 1: online, 2: disabled;</p>

## 2.13.2 Space Interface

### 2.13.2.1 Get Space List

<p><b>Post Request URL</b></p>	<p>http://serverIP:serverPort/api/space/list?access_token={apitoken}&amp;name={name}&amp;pageNo={pageNo}&amp;pageSize={pageSize}</p>
<p><b>Request Mode</b></p>	<p>GET</p>
<p><b>Request content</b></p>	<p>None</p>

<b>Request Parameter Description</b>	Name: space name;  pageNo: Page number, required, must be greater than 0;  pageSize: The number of each page,required, must be greater than 0, each page shall not exceed 1000;
<b>Response Results</b>	<pre> {   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 50,     "total": 1,     "data": [       {         "id": "8a8890e1936b8e2601936bb8d7380048",         "name": "test space1",         "picture": "",         "status": "0",         "capacity": "21",         "spaceFacilityIds": "8a8890e192b76ffd0192b771806537a6,8a8890e192b76ffd0192b771806637aa",         "spaceDeviceIds": "8a8890e192e039c60192e073913f4675",         "authAreaName": "Area Name",         "address": "D09 七楼 1011", </pre>

	<pre> "personId": "8a8890e1932832af019328c3ff422a94", "personPin": "3170079", "personTel": "151234567891", "smartSpace": true, "remark": "", "retentionTime": "15"     }   ],   "offset": 0,   "lastPage": true } </pre>
<p><b>Response Result Description</b></p>	<p>Data: array sets;</p> <p>Id: space id;</p> <p>Name: space name;</p> <p>Picture: space photo base64 string</p> <p>Status: status, 0: space, 1: in using, 2: disabled;</p> <p>Capacity: capacity</p> <p>spaceFacilityIds: Facility ID, multiple IDs are separated by commas;</p> <p>spaceDeviceIds: deviceid, multiple IDs are separated by commas;</p> <p>authAreaName: area name;</p> <p>address: space address;</p> <p>personId: manager person id;</p>

	<p>personPin: manager person pin;</p> <p>personTel: manager person's contact information;</p> <p>smartSpace: whether it's smart space, true: yes, false: no(If the space is configured with smart IoT devices and needs to support scene linkage, you can set it as a smart space);</p> <p>remark: space description;</p> <p>retentionTime: retention duration, four options: 15, 30, 45, 60 minutes, if it is empty, no space retention duration is set;</p>
--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.13.2.2 Query Space Information Based on Space ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/space/getById?id={id}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	None
<b>Request Parameter Description</b>	Id: space id, required;
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "8a8890e1936b8e2601936bb8d7380048",     "name": "test space1",     "picture": ""   } }</pre>

	<pre> "status": "0",  "capacity": "21",    "spaceFacilityIds": "8a8890e192b76ffd0192b771806537a6,8a8890e192b76ffd0192b771806637aa",  "spaceDeviceIds": "8a8890e192e039c60192e073913f4675",  "authAreaName": "Area Name",  "address": "D09 7 floor 1011",  "personId": "8a8890e1932832af019328c3ff422a94",  "personPin": "3170079",  "personTel": "151234567891",  "smartSpace": true,  "remark": "",  "retentionTime": "15"  }  } </pre>
<p><b>Response Result Description</b></p>	<p>Data: space information;</p> <p>Id: space id;</p> <p>Name: space name;</p> <p>Picture: space photo base64 string</p> <p>Status: status, 0: space, 1: in using, 2: disabled;</p> <p>Capacity: capacity</p> <p>spaceFacilityIds: Facility ID, multiple IDs are separated by commas;</p> <p>spaceDeviceIds: deviceid, multiple IDs are separated by commas;</p>

	<p>authAreaName: area name;</p> <p>address: space address;</p> <p>personId: manager person id;</p> <p>personPin: manager person pin;</p> <p>personTel: manager person's contact information;</p> <p>smartSpace: whether it's smart space, true: yes, false: no(If the space is configured with smart IoT devices and needs to support scene linkage, you can set it as a smart space);</p> <p>remark: space description;</p> <p>retentionTime: retention duration, four options: 15, 30, 45, 60 minutes, if it is empty, no space retention duration is set;</p>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.13.2.3 Delete Space Based on Space ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/space/deleteById?&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	<pre>{   "id": "8a8890e1936b8e2601936bb8d7380048" }</pre>
<b>Request Parameter Description</b>	Id: space id, required;
<b>Response Results</b>	<pre>{   "code": 0,   "message": "Succeed: 1, Failed: 0.", }</pre>

	<pre>"data": [] }</pre>
<b>Response Result Description</b>	Description of the generic response parameters

### 2.13.2.4 Add/Edit Space

<b>Post Request URL</b>	http://serverIP:serverPort/api/space/addSpace?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	<pre>{   "address": "D09 7 floor 101",   "capacity": 20,   "id": "",   "name": "test space",   "personPin": 1234567,   "personTel": 15123456789,   "picture": "",   "remark": "this is a space",   "retentionTime": 15,   "smartSpace": false,   "spaceDeviceIds": "8a8890e192e039c60192e073913f4675",   "spaceFacilityIds": "8a8890e192b76ffd0192b771806637aa,8a8890e192b76ffd0192b771806537a6"</pre>

	<pre> }</pre>
<b>Request Parameter Description</b>	<p>Id: space id, required, empty when adding, not empty when editing;</p> <p>Name: space name, required;</p> <p>Picture: space photo base64 string</p> <p>Status: status, 0: space, 1: in using, 2: disabled;</p> <p>Capacity: capacity for the space, required;</p> <p>spaceFacilityIds: Facility ID, multiple IDs are separated by commas;</p> <p>spaceDeviceIds: deviceid, multiple IDs are separated by commas, required;</p> <p>address: space address;</p> <p>personPin: manager person pin, required;</p> <p>personTel: manager person's contact information;</p> <p>smartSpace: whether it's smart space, true: yes, false: no(If the space is configured with smart IoT devices and needs to support scene linkage, you can set it as a smart space);</p> <p>remark: space description;</p> <p>retentionTime: retention duration, four options: 15, 30, 45, 60 minutes, if it is empty, no space retention duration is set;</p>
<b>Response Results</b>	<pre> {   "code": 0,   "message": "success",   "data": null }</pre>
<b>Response Result</b>	<p>Description of the generic response parameters</p>

<b>Description</b>	
--------------------	--

## 2.13.3 Space Facility Interface

### 2.13.3.1 Get Space Facility List

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceFacility/list?access_token={apitoken}&name={name}&pageNo={pageNo}&pageSize={pageSize}
<b>Request Mode</b>	GET
<b>Request content</b>	None
<b>Request Parameter Description</b>	<p>Name: space facility name;</p> <p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize: The number of each page, required, must be greater than 0, each page shall not exceed 1000;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 50,     "total": 3,     "data": [       {         "id": "8a8890e19371ca56019371deb3850010",</pre>

```
"name": "test facility1",  
  
"facilityNo": 22,  
  
"remark": "this is a facility11",  
  
"initFlag": false  
  
},  
  
{  
  
"id": "8a8890e192b76ffd0192b771806637aa",  
  
"name": "Projector",  
  
"facilityNo": 1,  
  
"remark": "",  
  
"initFlag": true  
  
},  
  
{  
  
"id": "8a8890e192b76ffd0192b771806537a6",  
  
"name": "Electronic Whiteboard",  
  
"facilityNo": 2,  
  
"remark": "",  
  
"initFlag": true  
  
}  
  
],  
  
"offset": 0,  
  
"lastPage": true  
  
}
```

	}
<b>Response Result Description</b>	Data: array sets; Id: facility id; Name: facility name; facilityNo: facility number; remark: facility description; initFlag: Whether to initialize data, true: initialize data, false: not initialize data;

### 2.13.3.2 Query Facility Information Based on Facility ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceFacility/getById?id={id}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	None
<b>Request Parameter Description</b>	Id: facility id, required;
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "8a8890e19371ca56019371deb3850010",     "name": "test facility1",     "facilityNo": 22,</pre>

	<pre> "remark": "this is a facility11",  "initFlag": false  }  } </pre>
<b>Response Result Description</b>	<p>Data: facility information;</p> <p>Id: facility id;</p> <p>Name: facility name;</p> <p>facilityNo: facility number;</p> <p>Remark: facility description;</p> <p>initFlag: Whether to initialize data;</p>

### 2.13.3.3 Delete Facility Based on Facility ID

<b>Post Request URL</b>	<p>http://serverIP:serverPort/api/spaceFacility/deleteById?id={id}&amp;access_token={apitoken}</p>
<b>Request Mode</b>	<p>POST</p>
<b>Request content</b>	<pre> {  "Id": "8a8890e19371ca56019371d7d0020009"  } </pre>
<b>Request Parameter Description</b>	<p>Id: facility id, required;</p>
<b>Response Results</b>	<pre> {  "code": 0,  "message": "Succeed: 1, Failed: 0.",  } </pre>

	<pre>"data": [] }</pre>
<b>Response Result Description</b>	Description of the generic response parameters

### 2.13.3.4 Add/Edit Space Facility

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceFacility/addFacility?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	<pre>{   "facilityNo": 20,   "id": "",   "name": "test facility",   "remark": "this is a facility" }</pre>
<b>Request Parameter Description</b>	<p>Id: facility id, required, empty when adding, not empty when editing;</p> <p>Name: facility name, required;</p> <p>facilityNo: facility number, required, integer greater than 20;</p> <p>remark: facility description;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": null }</pre>

	}
<b>Response Result Description</b>	Description of the generic response parameters

## 2.13.4 Space Service Interface

### 2.13.4.1 Get Space Service List

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceService/list?access_token={apitoken}&name={name}&pageNo={pageNo}&pageSize={pageSize}
<b>Request Mode</b>	GET
<b>Request content</b>	None
<b>Request Parameter Description</b>	Name: space service name; pageNo: Page number, required, must be greater than 0; pageSize: The number of each page, required, must be greater than 0, each page shall not exceed 1000;
<b>Response Results</b>	{ "code": 0, "message": "success", "data": { "page": 0, "size": 50, "total": 3, }

```
"data": [  
  {  
    "id": "8a8890e19370e2c70193715870d10026",  
    "name": "test service",  
    "remark": "this is a service1",  
    "initFlag": false  
  },  
  {  
    "id": "8a8890e192b76ffd0192b77180423758",  
    "name": "Cleaning",  
    "remark": "",  
    "initFlag": true  
  },  
  {  
    "id": "8a8890e192b76ffd0192b77180413754",  
    "name": "Snack",  
    "remark": "",  
    "initFlag": true  
  }  
],  
"offset": 0,  
"lastPage": true  
}
```

	}
<b>Response Result Description</b>	Data: array sets; Id: service id; Name: service name; remark: service description; initFlag: Whether to initialize data, true: initialize data, false: not initialize data;

### 2.13.4.2 Query Facility Information Based on Facility ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceService/getById?id={id}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	None
<b>Request Parameter Description</b>	Id: service id, required;
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "8a8890e19370e2c70193715870d10026",     "name": "test service ",     "remark": "this is a service1",     "initFlag": false   } }</pre>

	<pre> } } </pre>
<b>Response Result Description</b>	Data: service information; Id: service id; Name: service name; Remark: service description; initFlag: Whether to initialize data;

### 2.13.4.3 Delete Service Based on Service ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceService/deleteById?id={id}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	<pre> {   "id": "8a8890e19370e2c70193715870d10026" } </pre>
<b>Request Parameter Description</b>	Id: service id, required;
<b>Response Results</b>	<pre> {   "code": 0,   "message": "Succeed: 1, Failed: 0.",   "data": [] } </pre>
<b>Response</b>	Description of the generic response parameters

<b>Result Description</b>	
---------------------------	--

#### 2.13.4.4 Add/Edit Space Service

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceService/addService?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	<pre>{   "id": "",   "name": "test service",   "remark": "this is a service" }</pre>
<b>Request Parameter Description</b>	<p>Id: service id, required, empty when adding, not empty when editing;</p> <p>Name: service name, required;</p> <p>remark: facility description;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": null }</pre>
<b>Response Result Description</b>	Description of the generic response parameters

## 2.13.5 Space Reservation Interface

### 2.13.5.1 Get Space Reservation List

<b>Post Request URL</b>	http://serverIP:serverPort /api/spaceReservation/list?access_token={apitoken}&spaceName={spaceName}&subject={subject}&startTime={startTime}&endTime={endTime}&pageNo={pageNo}&pageSize={pageSize}
<b>Request Mode</b>	GET
<b>Request content</b>	None
<b>Request Parameter Description</b>	<p>spaceName: space name;</p> <p>Subject: subject name;</p> <p>startTime: time range query start time;</p> <p>endTime: Time range query end time;</p> <p>pageNo: Page number, required, must be greater than 0;</p> <p>pageSize: The number of each page,required, must be greater than 0, each page shall not exceed 1000;</p>
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "page": 0,     "size": 50,     "total": 1,     "data": [</pre>

```
{
  "id": "8a8890e19376a2ba019376a972860014",
  "subject": "gfdgfd",
  "content": "",
  "spaceId": "8a8890e192d0aeec0192dc8158d66e4f",
  "spaceName": "空间 1",
  "spaceAddress": "",
  "startSignInDateTime": "2024-11-29 15:44:00",
  "startTime": "2024-11-29 15:44:00",
  "endTime": "2024-11-29 16:44:00",
  "startSignInTime": 0,
  "applyPersonId": "8a8890e1932832af019328c3ff422a94",
  "applyPersonPin": "317009",
  "applyPersonName": "xxxxx",
  "applyPersonLastName": "lee",
  "applyPersonEmail": "rwq@zzz",
  "serviceIds": "",
  "reservationStatus": 3,
  "remindTime": "5m"
},
],
"offset": 0,
"lastPage": true
```

	<pre> } } </pre>
<p><b>Response Result Description</b></p>	<p>Data: array sets;</p> <p>subject: subject;</p> <p>content: reservation content;</p> <p>spaceId: space id;</p> <p>spaceName: space name;</p> <p>spaceAddress: space address;</p> <p>startSignInDateTime: start sign-in time;</p> <p>startTime: reservation start time;</p> <p>endTime: reservation end time;</p> <p>startSignInTime: how many minutes in advance to sign in, four options: 0, 5, 15, 30 (unit: minutes);</p> <p>applyPersonId: apply person id;</p> <p>applyPersonPin: apply person pin;</p> <p>applyPersonName: apply person name;</p> <p>applyPersonLastName: apply person's last name;</p> <p>applyPersonEmail: apply person's email;</p> <p>serviceIds: space service ID, multiple service IDs are separated by commas;</p> <p>reservationStatus: reservation status, 0: to be started, 1: in progress, 2: ended, 3: cancelled;</p> <p>remindTime: notification reminder time, 5m: 5 minutes in advance, 15m: 15 minutes in advance, 30m: 30 minutes in advance, 1h: 1 hour in advance, 1d: 1 day in advance;</p>

### 2.13.5.2 Query Reservation Information Based on Reservation ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceReservation/getById?id={id}&access_token={apitoken}
<b>Request Mode</b>	GET
<b>Request content</b>	None
<b>Request Parameter Description</b>	Id: reservation id, required;
<b>Response Results</b>	<pre>{   "code": 0,   "message": "success",   "data": {     "id": "8a8890e19376a2ba019376a972860014",     "subject": "gfdgfd",     "content": "",     "spaceId": "8a8890e192d0aeec0192dc8158d66e4f",     "spaceName": "space1",     "spaceAddress": "",     "startSignInDateTime": "2024-11-29 15:44:00",     "startTime": "2024-11-29 15:44:00",     "endTime": "2024-11-29 16:44:00",     "startSignInTime": 0,     "applyPersonId": "8a8890e1932832af019328c3ff422a94",   } }</pre>

	<pre> "applyPersonPin": "3170079",  "applyPersonName": "jeiffu",  "applyPersonLastName": "lee",  "applyPersonEmail": "rwq@zzz",  "participantIds": null,  "serviceIds": "",  "reservationStatus": 3,  "remindTime": "5m"  } </pre>
<b>Response Result Description</b>	<p>Data: reservation information;</p> <p>subject: subject;</p> <p>content: reservation content;</p> <p>spaceId: space id;</p> <p>spaceName: space name;</p> <p>spaceAddress: space address;</p> <p>startSignInDateTime: start sign-in time;</p> <p>startTime: reservation start time;</p> <p>endTime: reservation end time;</p> <p>startSignInTime: how many minutes in advance to sign in, four options: 0, 5, 15, 30 (unit: minutes);</p> <p>applyPersonId: apply person id;</p> <p>applyPersonPin: apply person pin;</p>

	<p>applyPersonName: apply person name;</p> <p>applyPersonLastName: apply person's last name;</p> <p>applyPersonEmail: apply person's email;</p> <p>serviceIds: space service ID, multiple service IDs are separated by commas;</p> <p>reservationStatus: reservation status, 0: to be started, 1: in progress, 2: ended, 3: cancelled;</p> <p>remindTime: notification reminder time, 5m: 5 minutes in advance, 15m: 15 minutes in advance, 30m: 30 minutes in advance, 1h: 1 hour in advance, 1d: 1 day in advance;</p>
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.13.5.3 Delete Reservation Based on Reservation ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceReservation/deleteById?id={id}&access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	<pre>{   "id": "8a8890e19376a2ba019376a972860014" }</pre>
<b>Request Parameter Description</b>	Id: reservation id, required;
<b>Response Results</b>	<pre>{   "code": 0,   "message": "Succeed: 1, Failed: 0.",   "data": [] }</pre>
<b>Response</b>	Description of the generic response parameters

<b>Result Description</b>	
---------------------------	--

#### 2.13.5.4 Cancel Reservation Based on Reservation ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceReservation/cancelReservation?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	{ "id": "8a8890e19399fa3701939a02f33a0015" }
<b>Request Parameter Description</b>	Id: reservation id, required;
<b>Response Results</b>	{ "code": 0, "message": "Succeed: 1, Failed: 0.", "data": [] }
<b>Response Result Description</b>	Description of the generic response parameters

#### 2.13.5.5 Finish Reservation Based on Reservation ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceReservation/finishReservation?access_token={apitoken}
-------------------------	-------------------------------------------------------------------------------------------

<b>Request Mode</b>	POST
<b>Request content</b>	<pre>{   "id": "8a8890e19399fa3701939a02f33a0015" }</pre>
<b>Request Parameter Description</b>	Id: reservation id, required;
<b>Response Results</b>	<pre>{   "code": 0,   "message": "Succeed: 1, Failed: 0.",   "data": [] }</pre>
<b>Response Result Description</b>	Description of the generic response parameters

### 2.13.5.6 Query Reservation Participants Based on Reservation ID

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceReservation/getReservationParticipants?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	<pre>{   "id": "8a8890e19399fa3701939a02f33a0015" }</pre>

<b>Request Parameter Description</b>	Id: reservation id, required;
<b>Response Results</b>	<pre> {   "code": 0,   "message": "success",   "data": [     {       "id": "8a8890e1932832af019328c3ff422a94",       "pin": "317009",       "name": "edf",       "lastName": "lee"     },     {       "id": "8a8890e1932832af0193288f0d531666",       "pin": "5",       "name": "",       "lastName": ""     }   ] } </pre>
<b>Response Result Description</b>	Data: reservation participant array list; Id: person id;

	Pin: person pin;
	Name: person name;
	lastName: person last name;

### 2.13.5.7 Add/Edit Space Reservation

<b>Post Request URL</b>	http://serverIP:serverPort/api/spaceReservation/addSpaceReservation?access_token={apitoken}
<b>Request Mode</b>	POST
<b>Request content</b>	<pre>{   "id": "",   "subject": "subject name",   "applyPersonEmail": "xxxx@zkteco.com",   "applyPersonPin": 1234567,   "content": "Please enter the content",   "endTime": "2024-12-06 18:50:19",   "participantIds": "8a8890e1932832af0193288ed42f162a,8a8890e1932832af0193288e1251634",   "remindTime": "15m",   "serviceIds": "8a8890e192b76ffd0192b7718036373d,8a8890e192b76ffd0192b771803e374e",   "spaceId": "8a8890e193956b62019395b01162016f",   "startSignInTime": 15,   "startTime": "2024-12-06 17:50:19", }</pre>

<p><b>Request Parameter Description</b></p>	<p>Id: reservation id, empty when adding, not empty when editing;</p> <p>content: content;</p> <p>applyPersonPin: apply person pin(required);</p> <p>applyPersonEmail: apply person's email;</p> <p>startTime: reservation start time;</p> <p>endTime: reservation end time;</p> <p>startSignInTime: how many minutes in advance to sign in, four options: 0, 5, 15, 30 (unit: minutes);</p> <p>participantIds: participant ID(required), multiple IDs are separated by commas;</p> <p>spcaeld: reservation space id(required);</p> <p>serviceIds: space service ID, multiple IDs are separated by commas;</p> <p>remindTime: notification reminder time, 5m: 5 minutes in advance, 15m: 15 minutes in advance, 30m: 30 minutes in advance, 1h: 1 hour in advance, 1d: 1 day in advance;</p>
<p><b>Response Results</b></p>	<pre>{   "code": 0,   "message": "success",   "data": null }</pre>
<p><b>Response Result Description</b></p>	<p>Description of the generic response parameters</p>

## 3 Appendix

### 3.1 Error Code

-1	Program error
-10	The department number or name cannot be empty
-12	The department name has already existed
-13	The department number does not exist
-14	Cannot be set as parent department
-15	Cannot delete department
-20	Pin number cannot be empty
-22	The person does not exist
-23	The card number has been used
-24	The level group does not exist
-25	No person under the access level
-26	The password has been used
-27	Invalid personnel photo
-29	PIN exception
-31	The personnel fingerprint already existed!
-32	Input wrong validtype!
-33	Fingerprint template index range is from 0-9!
-34	Template ID not existed!
-40	Authorize access failure
-41	Delete access level failure
-42	Personnel ID or access level id is null
-43	Access level id can't be null

-44	Door id can't be null
-45	Door open duration between 1~254 seconds.
-46	Door name can't be null
-47	Door doesn't exist
-48	Device serial number can't be null
-70	Advertisement doesn't exist
-71	Didn't support this format
-80	Area didn't exist
-81	Pins not allowed to be null
-82	Pins length not allowed over 500
-83	Area ID can't be null
-90	PageNo or pageSize cannot be set less than or equal to 0
-91	PageSize is greater than 1000
-92	PageSize and pageNo cannot be empty
-100	Time format error
-101	Start time cannot be greater than end time
-102	Device does not exist
-103	Device is offline or disabled
-104	Device does not support this feature
-105	Synchronization failed
-200	SN is required
-201	Floor Id is required
-202	The door opening time is required
-203	The door opening time needs to be greater than 0
-220	Operation type error

-221	Black and white list type error
-222	Time parameter error
-223	Car license plate is incorrect
-240	Host(visitor) pin cannot be empty
-241	Visitor certificate type cannot be empty
-242	Visitor certificate number cannot be empty
-243	Visitor name cannot be empty
-244	You have made a reservation
-245	No need to make a reservation during the visit
-246	Visit reason cannot be empty
-247	Visit date cannot be empty
-248	No eligible visitor reservation information
-249	Date format error
-250	Invalid certificate type
-251	Please fill in the correct national ID number
-252	Please fill in the correct date
-253	The employee does not exist
-254	User in the block list
-255	Registration has been completed. Please do not register again
-256	Visitor unregistered, cannot be sign-out
-257	Visitor information doesn't exist
-258	Visitor Number Count Must bigger than 0
-259	The operation object is not selected
-260	The visited department info doesn't exist
-261	Contains special characters

-262	The photo is not qualified.
-263	Reader ID cannot empty
-264	Did not exist this reader
-265	The Open Gate Type cannot empty
-266	Open Gate Type incorrect
-5001	Picture resolution is below 80000 pixels
-5002	No face was detected
-5003	Multiple human faces were detected
-5005	The face ratio is too small
-5006	Pictures are non-color images
-5007 -5008 -5016	The picture is vague
-5009	The picture was exposed seriously
-5010	The picture brightness is too dark
-5011	The picture is high noise
-5012	Face stretching too much
-5013	The face is blocked
-5014	Smile too much
-5015	The face deflection Angle is too large
-5017	Image brightness critical
-5018	Face deflection angle is critical

ZKTeco Industrial Park, No. 32, Industrial Road,  
Tangxia Town, Dongguan, China.

Phone : +86 769 - 82109991

Fax : +86 755 - 89602394

[www.zkteco.com](http://www.zkteco.com)



Copyright © 2025 ZKTECO CO., LTD. All Rights Reserved.