

USER MANUAL

ZKBio CVSecurity 3rd Party API

Version: 1.2

Date: August, 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	11
1.1 Authorize API Level	11
1.2 Interface Parameter	12
2. Interface Details	13
2.1 Access Control Interface	13
2.1.1 Delete Person Level [accLevel/deleteLevel]	13
2.1.2 Get Level by Id [accLevel/get byId{id}]	13
2.1.3 Get Level by Id [accLevel/get byId{id}]--V2	14
2.1.4 Get Level by Name [accLevel/get ByName{name}]	14
2.1.5 Get Level by Name [accLevel/get ByName{name}]--V2	15
2.1.6 Get Access Level/list [accLevel/list]	16
2.1.7 Sync Level Info to Device [accLevel/syncLevel]	17
2.1.8 Sync Person Level [accLevel/syncPerson]	17
2.1.9 Get Access Level/List (With total number)/[api/v2/accLevel/list]	18
2.1.10 Bulk Add/Edit Access Level Group[accLevel/addLevel]	19
2.1.11 Bulk Add/Edit Access Control Groups and Add Doors [accLevel/addLevelDoor]	20
2.1.12 Bulk Assign Access Level for Personnel [accLevel/addLevelPerson]	21
2.2 Advanced Access Control Interface	21
2.2.1 Retrieve Personnel by Area Code [accAdvanced/getWholsInsideByZone]	21
2.2.2 Retrieve a Collection of Personnel by Area Code [accAdvanced/getWholsInsideZoneList]	22
2.3 Elevator Control Level Interface	23
2.3.1 Delete Person Level [eleLevel/deleteLevel]	23
2.3.2 Get Ele Level by Id[eleLevel/getById{id}]	24
2.3.3 Get Ele Level by Name[eleLevel/getByName/{name}]	24
2.3.4 Get Ele Level Group List[[eleLevel/list]	25
2.3.5 Sync Ele Level Group Info to Device[eleLevel/syncLevel]	26
2.3.6 Sync Person Ele Level [eleLevel/syncPerson]	26
2.3.7 Get Elevator access level list(With total number) [/api/v2/eleLevel/list]	27

2.4 Park Authorization Interface	28
2.4.1 Authorize Park Black and White List [parkAuthorize/addBlackWhite]	28
2.4.2 Delete Park Black and White List[parkAuthorize/delBlackWhite]	28
2.4.3 Edit Park Black and White List[parkAuthorize/editBlackWhite]	29
2.5 Card Interface	29
2.5.1 Get Card List by Personnel ID [card/getCards/{pin}]	29
2.5.2 Get Card List by Personnel ID [card/getCards/{pin}]--V2	30
2.5.3 Set Person Card Info[card/set]	31
2.6 Department Interface	31
2.6.1 Add /Edit Department[department/add]	31
2.6.2 Delete Department by Code [department/delete/{code}]	32
2.6.3 Delete Department by Code[department/delete]--V2	32
2.6.4 Get Department Info by Code [department/get/{code}]	33
2.6.5 Get Department Info by Code [department/get/{code}]--V2	34
2.6.6 Get Department List [department/get?code]	34
2.7 Media Interface	36
2.7.1 Add or Update Media Info[adMedia/add]	36
2.7.2 Delete Media by Name[adMedia/delete/{name}]	36
2.7.3 Get Media Object by Name[adMedia/get/{name}]	37
2.8 Area Interface	38
2.8.1 Get Area Object by Area Code[area/get/{code}]	38
2.8.2 Get Area Object by Area Code[area/get/{code}] --V2	39
2.8.3 Get Area List [area/list]	39
2.8.4 Get Area List -Total Quantity[attAreaPerson/area/list]	40
2.9 Person Interface of Attendance Area	41
2.9.1 Delete Area Person [attAreaPerson/delete]	41
2.9.2 Set Area Person [attAreaPerson/set]	42
2.10 Access Control Device Interface	43
2.10.1 Get Device List [device/accList]	43

2.10.2 Get Device Info by SN [device/getAcc/{sn}]	44
2.10.3 Get Device List (With Total Number) [api/v2/device/list]	45
2.11 Elevator Device Interface	46
2.11.1 Get Ele Device List [eleDevice/eleList]	46
2.11.2 Get Ele Device Info by Sn [eleDevice/getEle/{sn}]	47
2.11.3 Get Ele Device List (With Total Number) [api/v2/eleDevice/eleList]	48
2.12 Door Interface	49
2.12.1 Get the State of All Doors [door/all DoorState]	49
2.12.2 Get Door State by Door Id [door/doorStateById]	52
2.12.3 Get Door State by the Sn of Device [door/doorStateBy{Sn}]	53
2.12.4 Get Door Info by Door id [door/get]	54
2.12.5 Get Door List [door/list]	55
2.12.6 Remote Close by Door Id [door/remoteCloseById]	56
2.12.7 Remote Close by Door Name [door/remoteCloseByName]	56
2.12.8 Remote Open by Door Id [door/remoteOpenById]	57
2.12.9 Remote Open by Door Name [door/remoteOpenByName]	57
2.12.10 Get Door List (With Total Number) [api/v2/door/list]	58
2.13 Floor Interface	59
2.13.1 Get Floor Info List [floor/list]	59
2.13.2 Remote Close by Floor Id [floor/remoteCloseById]	60
2.13.3 Remote Open by Floor Id [floor/remoteOpenById]	61
2.13.4 Get Floor Info List (With Total Number) [api/v2/floor/list]	61
2.14 Person Interface	63
2.14.1 Add/ Edit Person Info [person/add]	63
2.14.2 Delete Person Info by Pin [person/delete/{pin}]	64
2.14.3 Delete Person Info by Pin (With Total Number) [person/delete/{pin}]	64
2.14.4 Get Person Info by Pin [person/get/{pin}]Get person Info by Pin [person/get/{pin}]	65
2.14.5 Get Person Info by Pin (With Total Number) [person/get/{pin}]	66
2.14.6 Get Personnel Information according to the PIN Array and the Department Code Array	67

2.14.7 Get Dynamic QR Code by Personal Identification Number	69
2.14.8 Get Dynamic QR Code by Personal Identification Number--V2	69
2.14.9 Resignation	70
2.14.10 Add/Edit Personnel Basic Information	71
2.14.11 Update Personnel Photo by PIN	73
2.14.12 Get Personnel Information according to the PIN Array and the Department Code Array (With Total Number)	74
2.14.13 Reinstatement[api/person/reinstated]	77
2.14.14 Bulk Add/Edit Personnel Information [api/v2/person/addPersons]	80
2.14.15 Bulk Delete Personnel Information by PIN [api/v2/person/deleteByPins]	82
2.14.16 Detect whether a Facial Photo can be Extracted or Not [api/v2/person/detectFace] ..	83
2.15 Access Control Reader Interface	83
2.15.1 Get Access Control Reader Info List[reader/acclist]	83
2.15.2 Get Access Control Reader Info by Reader Id [reader/getAcc]	85
2.15.3 Get Access Control Reader Info List[api/v2/reader/list] (With Total Number)	85
2.16 Visitor Reservation Interface	87
2.16.1 Add Visitor Reservation[visReservation/add]	87
2.16.2 Delete Visitor Reservation[visReservation/del]	88
2.17 Visitor Level Interface	89
2.17.1 Get Visitor Level Group List[visLevel/list]	89
2.18 Visitor Registration Check Out Interface	90
2.18.1 Add Visitor Registration[visRegistration/add]	90
2.18.2 Visitor Check Out [visRegistration/exit]	91
2.18.3 Get Visitor Dynamic QR code	92
2.18.4 Get Visitor Dynamic QR code--V2	92
2.18.5 Retrieve a Dynamic QR Code Based on ID/Phone Number	93
2.18.6 Retrieve Visitor List	93
2.19 Access Control transaction Interface	95
2.19.1 Get Transaction by Device SN [transaction/device/{deviceSn}]	95
2.19.2 Get Transaction by Device SN (With Total Number) [v2/transaction/device/{deviceSn}]	97

2.19.3 Get Transaction by Device SN (With Total Number--V3) [v2/transaction/device/{deviceSn}]	100
2.19.4 Get Access Transaction List[transaction/list]	104
2.19.5 Get Access Transaction List (With Total Number)[api/v2/transaction/list]	106
2.19.6 Get Access Real-Time Monitoring Transaction[transaction/monitor]	109
2.19.7 Get Access Transaction by Pin [transaction/person/{pin}]	110
2.19.8 Get Access Transaction by Pin (With Total Number) [api/v2/transaction/person/{pin}]	112
2.19.9 Get Access Transaction List by pin--V3 (With Total Number) [api/v2/transaction/list]	115
2.19.10 Get Personnel First In and Last Out Record by Pin [transaction/firstInAndLastOut/{pin}]	118
2.19.11 Get Personnel First In and Last Out Record by Pin (With Total Number) [api/v2/transaction/firstInAndLastOut/{pin}]	119
2.19.12 Get Personnel First In and Last Out Record by Pin (With Total Number -V3) [api/v2/transaction/firstInAndLastOut/{pin}]	121
2.19.13 Obtain Door Transaction, Daily Transaction, Personnel Access Transactions [transaction/getDoorTransactions]	123
2.19.14 Obtain Door Transaction, Daily Transaction, Personnel Access Transactions (With Total Number) [api/v2/transaction/getDoorTransactions]	125
2.19.15 Get Doors Transaction Details [transaction/getDoorTransactionDetail]	129
2.19.16 Get Access Transactions by Event Id [api/transaction/getByld/{id}]	130
2.19.17 Get Access Transactions by Event Id (With Total Number) [api/transaction/getByld/{id}]	132
2.20 Elevator Transaction Interface	134
2.20.1 Get Elevator Transaction [eleTransaction/list]	134
2.20.2 Get Elevator Transaction (With Total Number) [api/v2/eleTransaction/list]	136
2.21 Park Transaction Interface	139
2.21.1 Get Park Entry Transaction [parkTransaction/listParkRecordin]	139
2.21.2 Get Park Exit Transaction[parkTransaction/listParkRecordout]	140
2.22 Att Transaction Interface	142
2.22.1 Get Att Transaction [transaction/listAttTransaction]	142
2.23 Att Device Interface	144
2.23.1 Get Att Device [device/getAtt]	144

2.24 Apply for Abnormal Attendance	145
2.24.1 Apply for Re-clock-in	145
2.24.2 Apply for Overtime	146
2.24.3 Ask for Leave	147
2.23.4 Get Leave Type List	148
2.23.5 Upload Leave Picture	149
2.23.6 Calculate the Total Duration of Leave	150
2.25 Edit Attendance Personnel Information	151
2.25.1 Edit Personal Information	151
2.26 Person BioTemplate Interface	152
2.26.1 Add /Edit Personnel BioTemplate	152
2.26.2 Retrieve BioTemplate Information by Personnel Number	153
2.26.3 Delete BioTemplate by Personnel ID	154
2.26.4 Delete Personnel BioTemplate Based on Personnel ID and BioTemplate ID	155
2.26.5 Retrieve BioTemplate Information by Personnel ID--V2	155
2.26.6 Delete BioTemplate Based on Personnel ID --V2	156
2.26.7 Delete Personnel BioTemplate Based on Personnel ID and BioTemplate ID --V2	157
2.27 Temperature Transactions Interface (Not Available in Some Versions)	157
2.27.1 Get Hep Transaction [api/hep/transaction/list]	157
2.27.2 Get Mask and Temperature Real-Time Monitor Events [api//hep/transaction/monitor]	160
2.28 FaceKiosk Personal Information	161
2.28.1 Edit Personal Information	161
2.29 FaceKiosk Area Personal	162
2.29.1 Delete Area Personnel	162
2.29.2 Add Area Personnel	163
2.29.3 Get a List of Attendance Zones	164
2.29.4 Get a list of attendance zones (With total number)	165
2.30 FaceKiosk Device Interface	166
2.30.1 Get Facekiosk Device List [api/device/attAdDeviceList]	166

2.30.2 Get Facekiosk Device List (With Total Number) [api/device/attAdDeviceList]	167
2.31 Entrance Control Device Interface	169
2.31.1 Get Entrance Control Device List [api/psgdDvice/list]	169
2.31.2 Get Entrance Control Device Info by SN [/api/psgDevice/getBySn]	170
2.31.3 Get Entrance Control Device List (With Total Number) [api/v2/psgDevice/list]	171
2.32 Entrance Control Gate Interface	172
2.32.1 Get the State of All Gates [psgGate/allGateState]	172
2.32.2 Get Entrance Control Gate State by Gate Id [psgGate/gateStateById]	173
2.32.3 Get Entrance Control Gate State by the SN of Device [psgGate/gateStateBySn]	174
2.32.4 Get Entrance Control Gate Info by Gate Id [psgGate/getById]	176
2.32.5 Get Entrance Control Gate List [api/psgGate/list]	176
2.32.6 Get Entrance Control Gate List (With Total Number) [api/v2/psgGate/list]	177
2.32.7 Remote Open by Gate Id [psgGate/remoteOpenById]	178
2.32.8 Remote Open by Gate Name [psgGate/remoteOpenByName]	179
2.33 Entrance Control Gate Interface	179
2.33.1 Delete Person Level [psgLevel/deleteLevel]	179
2.33.2 Get Gate Control Level by Id [psgLevel/getById/{id}]	180
2.33.3 Get Gate Control Level by Id [psgLevel/getById/{id}](v2)	180
2.33.4 Get Entrance Access Level by Name [psgLevel/getByName/{name}]	181
2.33.5 Get Entrance Access Level by Name [psgLevel/getByName/{name}](v2)	182
2.33.6 Get Entrance Control Access Level [/psgLevel/list]	182
2.33.7 Sync Level Info to Device [psgLevel/syncLevel]	183
2.33.8 Sync Person Level [/psgLevel/syncPerson]	183
2.33.9 Get Entrance Control Access Level (With Total Number) [api/v2/psgLevel/list]	184
2.33.10 Add Person Level By Level [api/v2/psgLevel/addLevelByLevel]	185
2.33.11 Add Person Level By Person [api/v2/psgLevel/addLevelByPerson]	186
2.34 Entrance Control Gate Reader Interface	187
2.34.1 Get Entrance Control Gate Reader Info List[psgReader/list]	187
2.34.2 Get Entrance Control Gate Reader Info List (With Total Number) [api/v2/psgReader/list]	188

2.34.3 Get Entrance Control Gate Reader Info by Reader Id [psgReader/getByld]	190
2.35 Entrance Control Device transaction Interface	191
2.35.1 Get Entrance Control Transaction by Device Sn [api/psgTransaction/device/{deviceSn}]	191
2.35.2 Get Entrance Control Transaction by Device Sn (With Total Number) [api/v2/psgTransaction/device/{deviceSn}]	193
2.35.3 Get Entrance Control Transaction by Device Sn (With Total Number-v3) [api/v3/psgTransaction/device]	195
2.35.4 Get Entrance Control Gate Transaction List[api/psgTransaction/list]	198
2.35.5 Get Entrance Control Gate Transaction (With Total Number)List[api/v2/psgTransaction/list]	199
2.35.6 Get Entrance Control Device Real-Time Monitoring Transaction [api/psgTransaction/monitor]	202
2.35.7 Get Personnel Entrance Control Access Transaction by Pin [api/psgTransaction/person/{pin}]	203
2.35.8 Get Entrance Control Gate Transaction List (With Total Number) [api/v2/psgTransaction/list]	205
2.35.9 Get Personnel Entrance Control Access Transaction by Pin (With Total Number) [api/v3/psgTransaction/person/{pin}]	207
2.36 Parking Device Interface	209
2.36.1 Retrieve the Device Channel List	209
2.36.2 The Specified Device Channel is Opened	210
2.36.3 New and Edit Parking Personnel License Plates	211
2.36.4 Delete the Parking Persons License Plates	211
2.37 Offline Consumption and Consumption Record Interface	212
2.37.1 Obtain the Offline Consumption Records	212
2.38 Online Consumption Account Interface	215
2.38.1 Get the Personnel Account Information through the Personnel Number	215
2.39 Online Consumption and Consumption Record Interface	217
2.39.1 Get Consumption Records	217
2.40 Smart Video Device Interface	221
2.40.1 Obtain the Main Device Interface	221
2.40.2 Get the Device Interface	223

2.40.3 Gets the Device Live Stream Address	225
2.40.4 Gets the Device Playback List	226
2.40.5 Obtain the Device Back to the Release Address	228
2.40.6 Set the Push Address	230
2.41 Parking Basic Management Interface	230
2.41.1 List of Parking Yard Areas (v2)	230
2.41.2 Get Parking Discount Strategy List (v2)	232
2.41.3 Parking Entry and Exit List (v2)	234
2.41.4 List of Parking Vehicle Types (v2)	235
2.41.5 Parking Space List (v2)	236
2.41.6 List of Parking Channels (v2)	238
2.41.7 Vehicle Admission List (v2)	239
2.41.8 Vehicle Exit List (v2)	241
2.41.9 Edit the Parking Allowed / Prohibited List (v2)	243
2.41.10 Parking of Authorized Vehicle List (v2)	244
2.41.11 Parking Delete Authorized Vehicle List (v2)	246
2.42 Space Device Interface	247
2.42.1 Get Space Device Info by Sn	247
2.42.2 Get Device List	248
2.43 Space Interface	250
2.43.1 Get Space List	250
2.43.2 Query Space Information Based on Space ID	252
2.43.3 Delete Space Based on Space ID	254
2.43.4 Add/Edit Space	255
2.44 Space Facility Interface	257
2.44.1 Get Space Facility List	257
2.44.2 Query Facility Information Based on Facility ID	259
2.44.3 Delete Facility Based on Facility ID	260
2.44.4 Add/Edit Space Facility	261

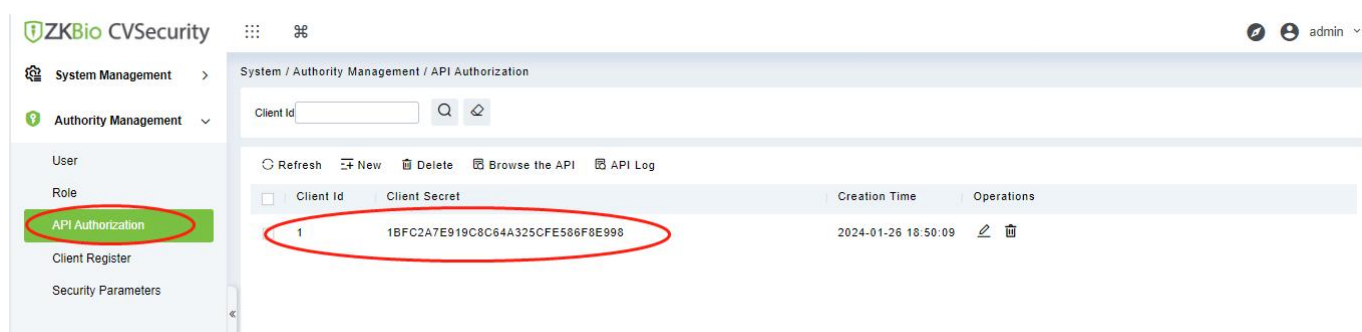
2.45 Space Service Interface	262
2.45.1 Get Space Service List	262
2.45.2 Query Facility Information Based on Facility ID	264
2.45.3 Delete Service Based on Service ID	265
2.45.4 Add/Edit Space Service	266
2.46 Space Reservation Interface	267
2.46.1 Get Space Reservation List	267
2.46.2 Query Reservation Information Based on Reservation ID	270
2.46.3 Delete Reservation Based on Reservation ID	272
2.46.4 Cancel Reservation Based on Reservation ID	273
2.46.5 Finish Reservation Based on Reservation ID	274
2.46.6 Query Reservation Participants Based on Reservation ID	275
2.46.7 Add/Edit Space Reservation	276
2.47 Parking Cost Interface	278
2.47.1 Parking Vehicle Billing	278
2.47.2 Parking Vehicle Payment Result Notification	279
3. Appendix	281
3.1 Error Code	281

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:



ZKBio CVSecurity		http://10.8.15.99:8098/api/api-docs	Explore
AccDevice : acc device	Show/Hide	List Operations	Expand Operations
AccDoor : acc door	Show/Hide	List Operations	Expand Operations
AccLevel : acc level	Show/Hide	List Operations	Expand Operations
AccReader : acc reader	Show/Hide	List Operations	Expand Operations
AccTransaction : acc transaction	Show/Hide	List Operations	Expand Operations
AdMedia : ins adMedia	Show/Hide	List Operations	Expand Operations
AttApply : att apply	Show/Hide	List Operations	Expand Operations
AttAreaPerson : att area person	Show/Hide	List Operations	Expand Operations
AttDevice : att device	Show/Hide	List Operations	Expand Operations
AttPerson : att person	Show/Hide	List Operations	Expand Operations
AttTransaction : att transaction	Show/Hide	List Operations	Expand Operations
EleDevice : ele device	Show/Hide	List Operations	Expand Operations
EleFloor : ele floor	Show/Hide	List Operations	Expand Operations

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

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

2. Interface Details

2.1 Access Control Interface

2.1.1 Delete Person Level [accLevel/deleteLevel]

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

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

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

	<pre>"name": "public level group" } }</pre>
Response Result Description	Data: level group object; Id: level group id; Name: level group name;

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

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

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

Post Request URL	http://serverIP:serverPort/api/accLevel/getByName/{name}?access_token={apitoken}
-------------------------	--

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

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

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

	}
Response Result Description	Data: level group object; Id: level group id, string; such as '402856aa6c3bbde1016c3bc2329200ff' Name: level group name;

2.1.6 Get Access Level/list [accLevel/list]

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

2.1.7 Sync Level Info to Device [accLevel/syncLevel]

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

2.1.8 Sync Person Level [accLevel/syncPerson]

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

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

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

2.1.10 Bulk Add/Edit Access Level Group[accLevel/addLevel]

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

2.1.11 Bulk Add/Edit Access Control Groups and Add Doors

[accLevel/addLevelDoor]

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

2.1.12 Bulk Assign Access Level for Personnel [accLevel/addLevelPerson]

Post Request URL	http://serverIP:serverPort/api/accLevel/addLevelPerson?levelIds={levelIds}&pin={pin}&access_token={apitoken}
Request Mode	POST
Request Content	None
Request Parameter Description	pin: Personnel ID, required; levelIds: Access group ID set, multiple IDs separated by commas, required.
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.2 Advanced Access Control Interface

2.2.1 Retrieve Personnel by Area Code [accAdvanced/getWholsInsideByZone]

Post Request URL	http://serverIP:serverPort/api/accAdvanced/getWholsInsideByZone?code={code}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	code: Area ID, required
Response Result	{ "code": 0, "message": "success", "data": [{

	<pre> "zoneId": "8a889006909539dd0190955d9a5a0fa8", "zoneName": "11", "pin": "20" }] } </pre>
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.2.2 Retrieve a Collection of Personnel by Area Code

[accAdvanced/getWholsInsideZoneList]

Post Request URL	http://serverIP:serverPort/api/accAdvanced/getWholsInsideByZoneList?pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
Request Mode	POST
Request Content	None
Request Parameter Description	PageNo: page no, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0;
Response Result	<pre> { "code": 0, "message": "success", "data": { "page": 0, "size": 2, "total": 5, "data": [</pre>

	<pre> { "zoneId": "8a888fe39319b9ee01931e657ac60c1e", "zoneName": "ZONE1", "pin": "10" }, { "zoneId": "8a888fe39319b9ee01931e657ac60c1e", "zoneName": "ZONE1", "pin": "77889" }], "offset": 0, "lastPage": false } </pre>
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.3 Elevator Control Level Interface

2.3.1 Delete Person Level [eleLevel/deleteLevel]

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

Response Result	Refer to public Response Result
Response Result Description	public response parameter description

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

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

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

Post Request URL	http://serverIP:serverPort/api/eleLevel/getByName/{name}?access_token={apitoken}
Request Mode	GET
Request Content	None

Request Parameter Description	name: level group name, required;
Response Result	<pre>{ "code": 0, "message": "success", "data": { "id": "402856aa6c3bbde1016c3bc2329200ff", "name": "public level group" } }</pre>
Response Result Description	data: level group object; id: level group id; name: level group name;

2.3.4 Get Ele Level Group List[[eleLevel/list]

Post Request URL	http://serverIP:serverPort/api/eleLevel/list?pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pageNo: page, required, must be greater than 0 pageSize: The number of per page, required, must be greater than 0
Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "id": "402856aa6c3bbde1016c3bc2329200ff", "name": "public level group" }] }</pre>

	}
Response Result Description	data: level group array list; id: level group id; name: level group name;

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

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

2.3.6 Sync Person Ele Level [eleLevel/syncPerson]

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

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

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

2.4 Park Authorization Interface

2.4.1 Authorize Park Black and White List [parkAuthorize/addBlackWhite]

Post Request URL	http://serverIP:serverPort/api/parkAuthorize/addBlackWhite?type={type}&carNumber={carNumber}&fromTime={fromTime}&toTime={toTime}&access_token={apitoken}
Request Mode	POST
Request Content	None
Request Parameter Description	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
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.4.2 Delete Park Black and White List[parkAuthorize/delBlackWhite]

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

2.4.3 Edit Park Black and White List[parkAuthorize/editBlackWhite]

Post Request URL	http://serverIP:serverPort/api/parkAuthorize/editBlackWhite?type={type}&carNumber={carNumber}&oldCarNumber={oldCarNumber}&fromTime={fromTime}&toTime={toTime}&access_token={apitoken}
Request Mode	POST
Request Content	None
Request Parameter Description	type: Black and White List type, 0: White List, 1:Black list carNumber: new license plate number, such as A32212 oldCarNumber: old license plate number, such as A32212 fromTime: start time, 2018-08-17(White List need to be filled in) toTime: end time 2019-08-16(White List need to be filled in)
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.5 Card Interface

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

Post Request URL	http://serverIP:serverPort/api/card/getCards/{pin}?access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pin: Personnel ID, required;
Response Result	<pre>{ "code": 0, "message": "success", "data": [</pre>

	<pre>{ "pin": "1234", "cardNo": "111", "cardType": "0" }</pre>
Response Result Description	data: card number List; pin: Personnel ID; cardNo: card no; 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

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

Post Request URL	http://serverIP:serverPort/api/card/getCards?pin={pin}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pin: Personnel ID, required;
Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "pin": "1234", "cardNo": "111", "cardType": "0" }] }</pre>

Response Result Description	data: card number List; pin: Personnel ID; cardNo: card no; 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
------------------------------------	---

2.5.3 Set Person Card Info[card/set]

Post Request URL	http://serverIP:serverPort/api/card/set?access_token={apitoken}
Request Mode	POST
Request Content	<pre>{ "pin": "1234", "cardNo": "212121", "cardType": "0" }</pre>
Request Parameter Description	Pin: pin, required cardNo: card number, required; cardType: card type, 0: master card, 1: sub card, the system does not open multi-card function, sub card setting invalid
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.6 Department Interface

2.6.1 Add /Edit Department[department/add]

Post Request URL	http://serverIP:serverPort/api/department/add?access_token={apitoken}
Request Mode	POST

Request Content	{ "name": "department name", "code": "456", "parentCode": "123" }
Request Parameter Description	Name: department name, required; code: department code, required; parentCode: parent department number;
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

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

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

2.6.3 Delete Department by Code[department/delete]--V2

Post Request URL	http://serverIP:serverPort/api/department/delete?code={code}&access_token={apitoken}
Request Mode	POST

Request Content	None
Request Parameter Description	code: department id, required;
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

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

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

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

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

2.6.6 Get Department List [department/get?code]

Post Request URL	http://serverIP:serverPort/api/department/getDepartmentList?pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
Request Mode	POST
Request Content	None
Request Parameter Description	pageNo: Page number, required, must be greater than 0. pageSize: Number of items per page, required, must be greater than 0.

Response Result

```
{  
  "code": 0,  
  "message": "success",  
  "data": {  
    "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> } } </pre>
Response Result Description	sortNo: Sorting number; name: Department name; code: Department code; parentCode: Parent department code.

2.7 Media Interface

2.7.1 Add or Update Media Info[adMedia/add]

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

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

Post Request URL	http://serverIP:serverPort/api/adMedia/delete/{name}&access_token={apitoken}
Request Mode	DELETE

Request Content	None
Request Parameter Description	Name: media name, required;
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

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

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

Response Result Description	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
------------------------------------	---

2.8 Area Interface

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

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

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

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

2.8.3 Get Area List [area/list]

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

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

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

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

2.9 Person Interface of Attendance Area

2.9.1 Delete Area Person [attAreaPerson/delete]

Post Request URL	http://serverIP:serverPort/api/attAreaPerson/delete?access_token={a pitoken}
Request Mode	DELETE

Request Content	{ "code": "1", "pins": "5" }
Request Parameter Description	Code: area code, required; Pins: pin, required
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.9.2 Set Area Person [attAreaPerson/set]

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

2.10 Access Control Device Interface

2.10.1 Get Device List [device/acclist]

Post Request URL	http://serverIP:serverPort/api/device/acclist?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pageNo: pageno, required, the value must be greater than 0; pageSize: the number of each page, required, must be greater than 0
Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "id": "402856aa6c3bbde1016c3bc2329200ff", "sn": "0564140100195", "name": "192.168.218.104", "type": "inBIO160", "state": "1", "module": "acc" }, { "id": "402856aa6c3bbde1016c3bc211e50059", "sn": "0566141900209", "name": "192.168.218.11", "type": "inBIO460", "state": "1", "module": "acc" }] }</pre>

	<pre>] }</pre>
Response Result Description	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;

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

Post Request URL	http://serverIP:serverPort/api/device/getAcc?sn={sn}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	Sn:device sn, required;
Response Result	<pre>{ "code": 0, "message": "success", "data": { "id": "402856aa6c3bbde1016c3bc2329200ff", "sn": "5662012052868", "name": "192.168.214.15", "type": "inibo460", "state": "0", "module": "acc" } }</pre>
Response Result Description	Data: device object; Id: device id; Sn: device sn

	Name: device name; Type: device model type; state: device state, 1 enabled, 0 disabled; Module: module, access, attendance, elevator;
--	--

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

Post Request URL	http://serverIP:serverPort/api/device/acclist?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pageNo: pageno, required, the value must be greater than 0; pageSize: the number of each page, required, must be greater than 0
Response Result	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 1, "total": 1, "data": [{ "id": "8a888e238478c3a6018478e40ee40a64", "sn": "CJI4200960003", "name": "10.8.14.142", "type": "SpeedFace-V5L", "status": "1", "module": "acc" }], "offset": 0, "lastPage": true } }</pre>

Response Result Description	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;
------------------------------------	---

2.11 Elevator Device Interface

2.11.1 Get Ele Device List [eleDevice/eleList]

Post Request URL	http://serverIP:serverPort/api/eleDevice/eleList?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pageNo: page no, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "id": "402856aa6c3bbde1016c3bc2329200ff", "sn": "0564140100195", "name": "192.168.218.104", "type": "EC-100", </pre>

	<pre> "state": "1", "module": "ele" }, { "id": "402856aa6c3bbde1016c3bc210d90016", "sn": "0566141900209", "name": "192.168.218.11", "type": "EC-100", "state": "1", "module": "ele" }] } </pre>
Response Result Description	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;

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

Post Request URL	http://serverIP:serverPort/api/eleDevice/getEle?sn={sn}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	Sn: device sn, required;
Response Result	<pre> { "code": 0, </pre>

	<pre> "message": "success", "data": { "id": "402856aa6c3bbde1016c3bc2329200ff", "sn": "5662012052868", "name": "192.168.214.15", "type": "EC-100", "state": "0", "module": "ele" } </pre>
Response Result Description	Data: device object; id: device id; sn: device sn; name: device name; type: device model name; state: state: device state, 1 enabled, 0 disabled; Module: module, access, attendance, elevator;

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

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

	<pre> "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>
Response Result Description	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;

2.12 Door Interface

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

Post Request URL	http://serverIP:serverPort/api/door/allDoorState?timestamp={timestamp}&access_token={apitoken}
-------------------------	---

Request Mode	GET
Request Content	None
Request Parameter Description	Timestamp: timestamp, millisecond, such as 1494467276539, required; Access frequency cannot be too high, it is recommended to obtain at least 10 seconds or more;
Response Result	<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", </pre>

	<pre> "name": "192.168.218.11-2", "deviceId": "402856aa6c3bbde1016c3bc211e5005a", "connect": "1", "sensor": "no sensor", "alarm": "none", "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>
Response Result Description	Data: door state array List; Id: door id; Name: door name; DeviceId: device id; Connection: connection state, 0 offline, 1 online, 2 disabled; Sensor: sensor description (unknown, no sensor, off, open) 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) relay: Relay state (off, open, unknown)
--	--

2.12.2 Get Door State by Door Id [door/doorStateById]

Post Request URL	http://serverIP:serverPort/api/door/doorStateById?doorId=1×tamp={timestamp}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	Timestamp: timestamp, millisecond, such as 1494467276539, required; Access frequency cannot be too high, it is recommended to obtain at least 10 seconds or more;
Response Result	<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>
Response Result Description	data: door state object id: door id; name: door name; deviceId: device id; connect: connection state (0 offline, 1 online, 2 disabled);

	sensor: sensor description (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)
--	---

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

Post Request URL	http://serverIP:serverPort/api/door/doorStateBySn?deviceSn={sn}×tamp={timestamp}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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
Response Result	<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>
Response Result Description	data: door state object; id: door id;

	name: door name; deviceId: device id; connect: device connection state (0 offline, 1 online, 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)
--	---

2.12.4 Get Door Info by Door id [door/get]

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

2.12.5 Get Door List [door/list]

Post Request URL	http://serverIP:serverPort/api/door/list?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	PageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
Response Result	<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>
Response Result Description	data: door array list; id: doorid; name: door name; deviceId: device id;

2.12.6 Remote Close by Door Id [door/remoteCloseById]

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

2.12.7 Remote Close by Door Name [door/remoteCloseByName]

Post Request URL	http://serverIP:serverPort/api/door/remotecloseByName?doorName={doorName}&access_token={apitoken}
Request Mode	POST
Request Content	None
Request Parameter Description	doorName: door name, required;
Response Result	Refer to the General Response Result Interface Parameter . For

	more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.12.8 Remote Open by Door Id [door/remoteOpenById]

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

2.12.9 Remote Open by Door Name[door/remoteOpenByName]

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

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

Post Request URL	http://serverIP:serverPort/api/v2/door/list?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	PageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
Response Result	<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>
Response Result Description	data: result; Page: current request page; Size: current data number of current request page; Total: total number; data: door array list; id: door id;

name: door name;
deviceId: device id;

2.13 Floor Interface

2.13.1 Get Floor Info List[floor/list]

Post Request URL	http://serverIP:serverPort/api/floor/list?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	PageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
Response Result	<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>
Response Result Description	data: floor array list; id: floor id; floorNo: floor no; name: floor name; deviceId: device id;

2.13.2 Remote Close by Floor Id[floor/remoteCloseById]

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

2.13.3 Remote Open by Floor Id[floor/remoteOpenById]

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

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

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

2.14 Person Interface

2.14.1 Add/ Edit Person Info [person/add]

Post Request URL	http://serverIP:serverPort/api/person/add?access_token={apitoken}
Request Mode	POST
Request Content	<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>
Request Parameter Description	Pin: pin, required deptCode: department code; name: name; lastName: person name (English) gender: gender, male, female; cardNo: card no; personPhoto: personnel comparison picture. "picture turn into base64 format" accLevelIds: access level groupid List, multiple comma separated, string; such as : 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;
Response Result	Refer to the General Response Result Interface Parameter . For more

	details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

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

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

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

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

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

Post Request URL	http://serverIP:serverPort/api/person/get/{pin}?access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pin: pin, required;
Response Result	<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>
Response Result Description	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;

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

Post Request URL	http://serverIP:serverPort/api/person/get?pin={pin}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pin: pin, required;
Response Result	<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>
Response Result Description	data: person data info; pin: personnel ID; gender: gender, M: male, F: female; deptCode: department code; deptName: department name; name: name;

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

2.14.6 Get Personnel Information according to the PIN Array and the Department Code Array

Post Request URL	http://serverIP:serverPort/api/person/getPersonList?access_token={apitoken}
Request Mode	POST
Request Content	<pre>{ "pins": "1,2,3", "deptCodes": "1,2", "pageNo": 1, "pageSize": 10 }</pre>
Request Parameter Description	pageNo: page number.(Page number is necessary and must be greater than zero.) pageSize: Number of entries per page is necessary and must be greater than zero.;
Response Result	<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>
Response Result Description	data: Personnel information date; pin: Personal identification number; gender: gender, M is male, F is female; deptCode: Department Code; deptName: Department Name; name: First Name; lastName: Last Name; cardNo: Card number; personPhoto: personal photo (base64 photo); vislightPhoto: Facial recognition photo (base64 photo); vislightPhotoPath: Facial recognition photo path;

2.14.7 Get Dynamic QR Code by Personal Identification Number

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

2.14.8 Get Dynamic QR Code by Personal Identification Number--V2

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

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

2.14.9 Resignation

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

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

2.14.10 Add/Edit Personnel Basic Information

Post Request URL	http://serverIP:serverPort/api/person/addPersonnelBasicInfo?access_token={apitoken}
Request Mode	POST
Request Content	<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>
Request Parameter Description	pin: Personal identification number and it is necessary; deptCode: Department Code; name: First Name; lastName: Last Name; gender: Gender, M: male, F: female; 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": birthday: Birthday; certNumber: Certificate number; 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; email: Email address, such as 123@zkteco.com; hireDate: employee's entry date, such as: " 2019-06-10";

	isDisabled: whether it is a prohibited list. True: Yes; False: No isSendMail: whether to send mail. True: send; False: do not send; mobilePhone: phone number,such as: "15123456789", personPwd: device authentication password such as: "123456"; ssn: social security number; 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";
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.14.11 Update Personnel Photo by PIN

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

Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

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

Post Request URL	http://serverIP:serverPort/api/v2/person/getPersonList?access_token={apitoken}
Request Mode	POST
Request Content	<pre>{ "pins": "1,2,3", "deptCodes": "1,2", "pageNo": 1, "pageSize": 10 }</pre>
Request Parameter Description	pageNo: page number.(Page number is necessary and must be greater than zero.) pageSize: Number of entries per page is necessary and must be greater than zero.
Response Result	<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>
Response Result Description	data: the result returned; page: The page number of the current request; size: The number of data requested by the current page number; total: total quantity; data: Personnel information date; pin: Personal identification number; gender: gender, M is male, F is female; deptCode: Department Code; deptName: Department Name; name: First Name; lastName: Last Name; cardNo: Card number;

	supplyCards: secondary cards; birthday: date of birth; selfPwd: self-service password; personPhoto: person's headshot base64 photo; vislightPhoto: comparison photo (base64 photo); vislightPhotoPath: path of the comparison photo; isSendMail: whether mailbox notification; mobilePhone: mobile phone number; personPwd: device authentication password; carPlate: car plate; email: email address; ssn : social security number; accLevelIds: access control privilege group id; accStartTime: access control privilege effective time from accEndTime: access control rights validity time Stop 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) certNumber: certificate number photoPath: photo path (base64) hireDate: entry time (2019-06-10) isDisabled: disabled list
--	---

2.14.13 Reinstatement[api/person/reinstated]

Post Request URL	http://serverIP:serverPort/api/person/reinstated?access_token={api token}
Request Mode	POST
Request Content	{ "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>
Request Parameter Description	pin: personnel number, required;

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";

	ssn: social security number; 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"; accLevellds: access control rights group id collection, more than one with comma separated, such as: 402856aa6c3c3c5063016c3cb3a0360005,402856aa6bff4b1a016bff5374800c0e,402856aa6c3baabb016c3bb37365000a .; accStartTime: access control effective start time, e.g. 2017-08-04 15:45:00; accEndTime: access control valid end time, e.g., 2017-08-04 23:59:59;
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

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

Post Request URL	http://serverIP:serverPort/api/v2/person/addPersons?access_token={apitoken}
Request Mode	POST
Request Content	<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"
```

	}]
Request Parameter Description	Pin: pin, required deptCode: department code; name: name; lastName: person name (English) gender: gender, male, female; cardNo: card no; personPhoto: personnel comparison picture. "picture turn into base64 format" accLevelIds: access level groupid List, multiple comma separated, string; such as : 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;
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.14.15 Bulk Delete Personnel Information by PIN

[api/v2/person/deleteByPins]

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

Response Result Description	Refer to General Response Result Parameter Description Interface Parameter
------------------------------------	--

2.14.16 Detect whether a Facial Photo can be Extracted or Not

[api/v2/person/detectFace]

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

2.15 Access Control Reader Interface

2.15.1 Get Access Control Reader Info List[reader/accList]

Post Request URL	http://serverIP:serverPort/api/reader/accList?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None

Request Parameter Description	PageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
Response Result	<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>
Response Result Description	Data: reader array List; Id: reader id; DoorId: door id; Name: reader name; ReaderNo: readerNo; ReaderState: reader state, 0: in, 1: out;

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

Post Request URL	http://serverIP:serverPort/api/reader/getAcc?id={id}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	id: reader id, required;
Response Result	<pre>{ "code": 0, "message": "success", "data": { "id": "402856aa6c3c5063016c3cb3a0360005", "doorId": "402856aa6bff4b1a016bff5374800c0e", "name": "192.168.214.15-1-in", "readerNo": 1, "readerState": 0 } }</pre>
Response Result Description	data: reader info object; id: reader id; doorId: doorid; name: reader name; readerNo: reader no; readState: reader state, 0: in, 1:out;

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

Post Request URL	http://serverIP:serverPort/api/v2/reader/list?pageNo=1&pageSize=2&access_token={apitoken}
-------------------------	---

Request Mode	GET
Request Content	None
Request Parameter Description	pageNo: page, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0
Response Result	<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", "readerNo": 2, "readerState": 1 }] } } </pre>
Response Result Description	Data: reader array List; page: the page number of the current request; size: the number of data requested by the current page number;

	total: total number; Id: reader id; DoorId: door id; Name: reader name; ReaderNo: readerNo; ReaderState: reader state, 0: in, 1: out;
--	--

2.16 Visitor Reservation Interface

2.16.1 Add Visitor Reservation[visReservation/add]

Post Request URL	http://serverIP:serverPort/api/visReservation/add?access_token={apitoken}
Request Mode	POST
Request Content	<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" }</pre>
Request Parameter Description	persPersonPin: interviewee pin, required; certType: certificate type, required; details: <pre>{ 2:national ID card; 3:Passport; 4:Driver's license; 5:Work Permit; 6:Real-name card; 7:driving license; 8:others; }</pre>

	certNum: certificate no, required; visitEmpName: interviewee name, required; visitEmpPhone: visitor phone; company: company; visitReason: visit reason, required; visitDate: visit date, yyyy-MM-dd mm:hh:ss, required; visitEndDate: visit date, yyyy-MM-dd mm:hh:ss, required;
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.16.2 Delete Visitor Reservation[visReservation/del]

Post Request URL	http://serverIP:serverPort/api/visReservation/del?certType={certType}&certNum={certNum}&visitDate={visitDate}&access_token={apitoken}
Request Mode	POST
Request Content	None
Request Parameter Description	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]
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result	Refer to General Response Result Parameter Description Interface

Description	Parameter
-------------	---------------------------

2.17 Visitor Level Interface

2.17.1 Get Visitor Level Group List[visLevel/list]

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

2.18 Visitor Registration Check Out Interface

2.18.1 Add Visitor Registration[visRegistration/add]

Post Request URL	http://serverIP:serverPort/api/visRegistration/add?&access_token={apitoken}
Request Mode	POST
Request Content	<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>
Request Parameter Description	"persPersonPin": "Interviewee pin", required certType: certificate type, required; details: <pre>{ 2: national ID card; 3: Passport; 4: Driver's license; 5: Work Permit; 6: Real-name card; 7: driving license; 8: others; }</pre> "certNum": visitor certificate no, required "visitEmpName": visitor name, required "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
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.18.2 Visitor Check Out [visRegistration/exit]

Post Request URL	http://serverIP:serverPort/api/visRegistration/exit?certType={certType}&certNum={certNum}&access_token={apitoken}
Request Mode	POST
Request Content	None
Request Parameter Description	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": visitor certificate no, required
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.18.3 Get Visitor Dynamic QR code

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

2.18.4 Get Visitor Dynamic QR code--V2

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

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

2.18.5 Retrieve a Dynamic QR Code Based on ID/Phone Number

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

2.18.6 Retrieve Visitor List

Post Request URL	http://serverIP:serverPort/api/visVisitor/getVisitorList?access_token={apitoken}&certNumbers={certNumbers}&names={names}&p
-------------------------	--

	ageNo={pageNo}&pageSize={pageSize}
Request Mode	POST
Request Content	None
Request Parameter Description	pageNo: Page number, required, must be grater than 0; pageSize: Page size, required, must be grater than 0, No more than 1000 items per page. certNumbers: Certification ID. names: Visitor Name
Response Result	<pre> { "code": 0, "message": "success", "data": [{ "id": "8a888ea18d584d12018d592429020014", "pin": "8000000001", "name": "123", "lastName": null, "certType": "8", "certNumber": "123", "company": "", "carPlate": "", "phone": "", "email": "", </pre>

	<pre> "status": null, "enabled": true, "nationality": null, "gender": "" }]</pre>
Response Result Description	

2.19 Access Control transaction Interface

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

Post Request URL	http://serverIP:serverPort/api/transaction/device/{deviceSn}?pageNo=1&pageSize=20&acc_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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;
Response Result	<pre> { "code": 0, "message": "success", "data": [{ "id": "402856aa6c3c5063016c3cb3a0360005", "eventTime": "2017-08-04 15:59:12", "pin": "", </pre>

	<pre> "name": "", "lastName": "", "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 }, { "id": "402856aa6c028430016c02d48ac10003", "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>
Response Result Description	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; 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;
--	---

2.19.2 Get Transaction by Device SN (With Total Number)

[v2/transaction/device/{deviceSn}]

Post Request URL	http://serverIP:serverPort/api/v2/transaction/device/{deviceSn}?pageNo=1&pageSize=20&acc_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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; startDate: startDate, [not required] endDate: endDate, [not required]
Response Result	{ "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": "",
    "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",

```

	<pre> "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, "eventLevel": 0, "doorName": "10.8.14.210-1", "capturePhotoBase64": "" }], "offset": 0, "lastPage": false } </pre>
Response Result Description	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; 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)
--	--

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

[v2/transaction/device/{deviceSn}]

Post Request URL	http://serverIP:serverPort/api/v3/transaction/device?deviceSn={deviceSn}&pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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; startDate: startDate, [not required] endDate: endDate, [not required]
Response Result	<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": "",
    "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",

```

	<pre> "capturePhotoBase64": "", "doorNumber": 1 }], "offset": 0, "lastPage": false } } </pre>
Response Result Description	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)
--	--

2.19.4 Get Access Transaction List[transaction/list]

Post Request URL	http://serverIP:serverPort/api/transaction/list?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	personPin: personPin, [not required] startDate: startDate, [not required] endDate: endDate, [not required] 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;
Response Result	<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": "" }] }</pre>

	<pre> "readerName": "others", "accZone": null, "devName": "192.168.214.15", "logId": 11734 }, { "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>
Response Result Description	data: transaction array list; id:5; 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; eventName: transaction description; eventPointName: transaction point name; readerName: reader name; accZone: access area name; devName: device name; logId: transaction index value, there will be a value if device support it;
--	--

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

Post Request URL	http://serverIP:serverPort/api/v2/transaction/list?personPin={personPin}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	personPin: personPin, [not required] startDate: start Date, [not required] endDate: end Date, [not required] 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;
Response Result	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 2, "total": 39, "data": [{</pre>

	<pre> "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", </pre>
--	--

	<pre> "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>
Response Result Description	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; 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)
--	---

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

Post Request URL	http://serverIP:serverPort/api/transaction/monitor?timestamp={timestamp}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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
Response Result	<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- out", "devName": "192.168.218.11(0566141900209)" }] }</pre>

	<pre>] }</pre>
Response Result Description	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;

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

Post Request URL	http://serverIP:serverPort/api/transaction/person/{pin}?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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]
Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "id": "402856aa6c17518e016c175385fd0001",</pre>

```

"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",
"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
}
]
}

```

Response Result Description	data:transaction array list; id:5; 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; eventName: transaction description; eventPointName: transaction point name; readerName: reader name; accZone: access area name; devName: device name; logId: transaction index value, there will be a value if device support it;
------------------------------------	---

2.19.8 Get Access Transaction by Pin (With Total Number)

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

Post Request URL	http://serverIP:serverPort/api/v2/transaction/person/{pin}?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	personPin: personPin, [not required] startDate: Start Date, [not required] endDate: End Date, [not required] 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;

Response Result

```
{
  "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": "",
      "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",
```

	<pre> "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, "eventLevel": 0, "doorName": "10.8.14.210-1", "capturePhotoBase64": "" }], "offset": 0, "lastPage": false } } </pre>
Response Result Description	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; 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)
--	--

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

[api/v2/transaction/list]

Post Request URL	http://serverIP:serverPort/api/v3/transaction/person?pageNo=1&pageSize=20&pin={pin}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	Pin:Person ID, [not required] startDate: startDate, [not required] endDate: endDate, [not required] 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;
Response Result	{ "code":0,

```
"message":"success",

"data":{

  "page":0,

  "size":1,

  "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,
```

	<pre> "eventNo":0, "eventLevel":0, "doorName":"10.8.14.210-1", "capturePhotoBase64":"" }], "offset":0, "lastPage":false } }</pre>
Response Result Description	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: personnel ID; 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; accZone: Access zone name

	devName: device name; logId: transaction index value, there will be a value if device support it; eventNo: Event ID eventLevel: Event Level doorName: Door Name capturePhotoBase64: Captured Photo (base64)
--	--

2.19.10 Get Personnel First In and Last Out Record by Pin

[transaction/firstInAndLastOut/{pin}]

Post Request URL	http://serverIP:serverPort/api/transaction/firstInAndLastOut/{pin}?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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]
Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "id": "40285ab374521d14017452820eb50005", "pin": "147", "name": "", "lastName": "" }] }</pre>

	<pre> "readerNameIn": "192.168.218.183-1-Out", "firstInTime": "2020-09-03 13:48:07", "readerNameOut": null, "lastOutTime": null, "deptCode": "1", "deptName": "Department Name" }] } </pre>
Response Result Description	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; readerNameOut: Last Out reader name; lastOutTime: Last Out Time;

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

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

Post Request URL	http://serverIP:serverPort/api/v2/transaction/firstInAndLastOut/{pin}?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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]
Response Result	<pre>{ "code": 0, "message": "success", "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>
Response Result Description	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; readerNameOut: Last Out reader name; lastOutTime: Last Out Time;
--	--

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

Post Request URL	http://serverIP:serverPort/api/v2/transaction/firstInAndLastOut?pageNo=1&pageSize=20&pin={pin}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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]
Response Result	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 2,</pre>

	<pre> "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>
Response Result Description	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;
--	---

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

Post Request URL	http://serverIP:serverPort/api/transaction/getDoorTransactions?access_token={apitoken}
Request Mode	POST
Request Content	<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>
Request Parameter Description	devsn: Device Serial Number [Not required] startTime: Start Time [Not required] endTime: End Time [Not required] filter: Filter conditions [Not required, filter by name, last name, PIN] (Fuzzy query) type: Event Type [Not required, alarm events, normal event, warning events] eventNo: Events Number [Not required]

	personPin: personnel ID number[Not required] [Accurate Query] page: page number pageSize: record number of current page
Response Result	<pre> { "ret": "ok", "msg": "operation succeed", "data": [{ "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, </pre>

	<pre> "page": 1, "pageSize": 10, "total": 2, "success": true } { "ret": "fail", "msg": "failed info", "data": null, "i18nArgs": null } </pre>
Response Result Description	data: transaction array list; Id: event id eventNo: event number eventName: event name eventTime: event time eventLevel: 0 normal 1 warning 2 alarm

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

Post Request URL	http://serverIP:serverPort/api/v2/transaction/getDoorTransactions?access_token={apitoken}
Request Mode	POST
Request Content	<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", </pre>

	<pre> "pageNo": "1", "pageSize": "2" } </pre>
Request Parameter Description	devsn: Device Serial Number [Not required] startTime: Start Time [Not required] endTime: End Time [Not required] filter: Filter conditions [Not required, filter by name, last name, PIN] (Fuzzy query) type: Event Type [Not required, alarm events, normal event, warning events] eventNo: Events Number [Not required] personPin: personnel ID number [Not required] [Accurate Query] page: page number pageSize: record number of current page
Response Result	<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", </pre>

```

"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": ""
},
{
"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

```

	<pre> }, "i18nArgs": null, "success": true } { "ret": "fail", "msg": "failed message", "data": null, "i18nArgs": null } </pre>
Response Result Description	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; 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)
--	---

2.19.15 Get Doors Transaction Details

[transaction/getDoorTransactionDetail]

Post Request URL	http://serverIP:serverPort/api/transaction/getDoorTransactionDetail?access_token={apitoken}
Request Mode	POST
Request Content	{ "id": "402856a7695c716301695c81290d0006", }
Request Parameter Description	{ "content": { id: event ID } }
Response Result	{ "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 } }

	<pre> "readerName": "others", reader name "verifyModeName": "others" Verification mode }, "i18nArgs": null, "success": true } { "ret": "fail", "msg": "failed info", "data": null, "i18nArgs": null } </pre>
Response Result Description	data:transaction array list; id: event id; eventNo: event number eventName: event name eventTime: event time

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

Post Request URL	http://serverIP:serverPort/api/transaction/getById/{id}?access_to ken={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	id: event id, required; eg: "8a888e8f83f376e50183f4019fb70bcf"
Response Result	<pre> { "code": 0, "message": "success", "data": { "id": "8a888e8f83f376e50183f4019fb70bcf", "eventTime": "2022-10-17 17:53:22", </pre>

	<pre> "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", "accZone": null, "devName": "10.8.14.181", "logId": 4521, "eventNo": 0, "eventLevel": 0, "doorName": "10.8.14.181-1", "capturePhotoBase64": "/9j/4AAQSkZJRgAB..." } }</pre>
Response Result Description	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; 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)
--	--

2.19.17 Get Access Transactions by Event Id (With Total Number)

[api/transaction/getById/{id}]

Post Request URL	http://serverIP:serverPort/api/v2/transaction/getById?id={id}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	id: event id, required; eg: "8a888e8f83f376e50183f4019fb70bcf"
Response Result	<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",</pre>

	<pre> "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, "eventNo": 0, "eventLevel": 0, "doorName": "10.8.14.181-1", "doorNumber": 1 "capturePhotoBase64": "/9j/4AAQSkZJRgAB..." } </pre>
Response Result Description	id: "402856aa6c032ca3016c03351d2600fd"; eventTime: record device trigger time; pin: personnel id; name: first 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; accZone: access area name devName: device name;

	logId: transaction index value, there will be a value if device support it; eventNo: event id eventLevel: event level, 0: normal, 1: abnormal, 2: Alarm doorName: door name capturePhotoBase64: captured photo of the door (base64)
--	---

2.20 Elevator Transaction Interface

2.20.1 Get Elevator Transaction [eleTransaction/list]

Post Request URL	http://serverIP:serverPort/api/eleTransaction/list?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	personPin: personPin, [not required] startDate: startDate, [not required] endDate: endDate, [not required] 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;
Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "id": "402856aa6c1778b3016c1a695c860013", "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", "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>
Response Result Description	data: transaction array list; id: 5; 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; 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;
--	--

2.20.2 Get Elevator Transaction (With Total Number)

[api/v2/eleTransaction/list]

Post Request URL	http://serverIP:serverPort/api/v2/eleTransaction/list?personPin={personPin}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	personPin: personPin, [not required] startDate: startDate, [not required] endDate: endDate, [not required] 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;
Response Result	<pre>{ "code": 0, "message": "success", "data": {</pre>

```

"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": "",
    "areaName": "Area",
    "cardNo": "",
    "devSn": "5662012052868",
    "verifyModeName": "Other",
    "eventName": "Connect to Server",
    "eventPointName": "",
  }
]

```

	<pre> "readerName": "Other", "devName": "192.168.214.15", "logId": 11733 }], "offset": 1, "lastPage": true } }</pre>
Response Result Description	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: event record array collection; id: "402856aa6c1778b3016c1a695c860016", eventTime: recording device trigger time; pin: personnel number; name: name of the personnel; lastName: the name of the personnel, only useful in English; deptName: department name; areaName: area name; cardNo: card number; devSn: device serial number; verifyModeName: verify mode name; eventName: event description; eventPointName: event departure point name; readerName: name of the reader head; devName: device name; logId: event index value, only have value according to whether the device supports it or not;

2.21 Park Transaction Interface

2.21.1 Get Park Entry Transaction [parkTransaction/listParkRecordin]

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

	<pre> "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>
Response Result Description	data: transaction array list; 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: transaction type; checkInTime: check-in time

2.21.2 Get Park Exit Transaction[parkTransaction/listParkRecordout]

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

	greater than 0, not more than 1000 per page;
Response Result	<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", "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>
Response Result Description	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
--	---

2.22 Att Transaction Interface

2.22.1 Get Att Transaction [transaction/listAttTransaction]

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

	<pre> "areaName": "areaname", "cardNo": "", "devSn": "5662012052868", "verifyModeName": "others", "eventName": "enable device", "eventPointName": "", "readerName": "others", "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>
Response Result Description	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; eventName: transaction description; eventPointName: transaction point name; readerName: reader name; accZone: access area name; devName: device name; logId: transaction index value, there will be a value if device support it;
--	---

2.23 Att Device Interface

2.23.1 Get Att Device [device/getAtt]

Post Request URL	http://serverIP:serverPort/api/device/getAtt?sn={sn}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	sn: Device Serial Number,[required]
Response Result	<pre>{ "code": 0, "message": "success", "data": { "id": "402856aa6c3bbde1016c3bc210d90016",</pre>

	<pre> "sn": "CEXB193160012", "name": "CEXB193160012", "type": "ZK-S1007", "status": "0", "module": "att" } } </pre>
Response Result Description	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;

2.24 Apply for Abnormal Attendance

2.24.1 Apply for Re-clock-in

Post Request URL	http://serverIP:serverPort/api/attApply/applySign?access_token={apitoken}
Request Mode	POST
Request Content	<pre> { "personPin": 1, "remark": "api", "signDatetime": "2021-11-19 18:00:00" } </pre>
Request Parameter	personPin (string): personal number

Description	remark (string): remarks signDatetime(string): Re-clock-in time
Response Result	{ "code":0, "message":"success", "data":null }
Response Result Description	

2.24.2 Apply for Overtime

Post Request URL	http://serverIP:serverPort/api/attApply/applyOvertime?access_token={apitoken}
Request Mode	POST
Request Content	{ "endDatetime": "2021-11-19 18:00:00", "personPin": 1, "remark": "api", "startDatetime": "2021-11-19 13:00:00" }
Request Parameter Description	endDatetime (string): End Time personPin (string): Personal Number remark (string): Remarks startDatetime (string): start time

Response Result	<pre>{ "code":0, "message":"success", "data":null }</pre>
Response Result Description	

2.24.3 Ask for Leave

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

	leaveTypeNo (string): Leave type number(For details/api/attApply/getLeaveTypeList) personPin (string): Personal number remark (string): Remark startDatetime (string): Start time
Response Result	<pre>{ "code":0, "message":"success", "data":null }</pre>
Response Result Description	

2.23.4 Get Leave Type List

Post Request URL	http://serverIP:serverPort/api/attApply/getLeaveTypeList?access_token={apitoken}
Request Mode	GET
Request Content	
Request Parameter Description	
Response Result	<pre>{ "code": 0, "message": "success", "data": [{</pre>

	<pre> "leaveTypeNo": "L1", "leaveTypeName": "personal leave" }, { "leaveTypeNo": "L2", "leaveTypeName": "marriage leave" }] }</pre>
Response Result Description	leaveTypeNo: Leave type number leaveTypeName: Leave type name

2.23.5 Upload Leave Picture

Post Request URL	http://serverIP:serverPort/api/attApply/uploadLeavelImage?accesstoken={apitoken}
Request Mode	POST
Request Content	<pre> { "base64Image": "base64Image" }</pre>
Request Parameter Description	base64Image (string): picture Base64code
Response Result	<pre> { "code":0, "message":"success", }</pre>

	<pre>"data":"/fileresource/att/leave/image/2021-12-03/1638503394663.jpg" }</pre>
Response Result Description	data: picture path

2.23.6 Calculate the Total Duration of Leave

Post Request URL	http://serverIP:serverPort/api/attApply/getApplyTimeLong?access_token={apitoken}
Request Mode	POST
Request Content	<pre>{ "endDatetime": "2021-11-19 18:00:00", "leavelImagePath": "string", "leaveTypeNo": "L1", "personPin": 1, "remark": "api", "startDatetime": "2021-11-19 13:00:00" }</pre>
Request Parameter Description	endDatetime (string): End time leavelImagePath (string, optional): Leave photo path (For details/api/attApply/uploadLeavelImage) leaveTypeNo (string): Leave type number (For details/api/attApply/getLeaveTypeList)

	personPin (string): personal number remark (string,): Remark startDatetime (string): Start time
Response Result	{ "code": 0, "message": "success", "data": null }
Response Result Description	

2.25 Edit Attendance Personnel Information

2.25.1 Edit Personal Information

Post Request URL	http://serverIP:serverPort /api/attPerson/editAttPerson?access_token={apitoken}
Request Mode	POST
Request Content	<pre>{ "isAttendance": false, "perDevAuth": 0, "personPin": 1, "verifyMode": 0 }</pre>
Request Parameter Description	isAttendance (boolean): Attendance or not (true is Yes, false is Not) perDevAuth (integer): Device operation authority (0 is ordinary person, 2 is registrar, 6

	is administrators, 14 is super administrators) personPin (string): personal number 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 prints and fingerprint , 29=Palm prints and fingerprint and face)
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.26 Person BioTemplate Interface

2.26.1 Add /Edit Personnel BioTemplate

Post Request URL	http://serverIP:serverPort/api/bioTemplate/add?access_token={apitoken}
Request Mode	Post
Request Content	{

	<pre> "personId": "666", "validType": "1", "template": "fsfsfsfsfsfsfsfsfsfsfsfsfsfs", "templateNo": "3" } </pre>
Request Parameter Description	personId: personnel ID, required; validType: 1:common, 3:duress, required; Template: personnel fingerprint template, required; templateNo: finger number, required,range[0-9];
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.26.2 Retrieve BioTemplate Information by Personnel Number

Post Request URL	http://serverIP:serverPort/api/bioTemplate/getFgListByPin/{pin} ?access_token={apitoken}
Request Mode	GET
Request Content	NONE
Request Parameter Description	Pin: personnel ID, required;
Response Result	<pre> { "code": 0, "message": "success", "data": { "id": "40280bb470e118510170e13f57dc001a", "pin": "3170156", "validType": "1", </pre>

	<pre> "bioType": 1, "version": "10", "template": "TWITUzlxAAAkEikECAUHCc7QAAAcK2kBAAA AhNcowCrDALI P3.....(Biotemplate)", "templateNo": "6", "templateNoIndex": 0 } }</pre>
Response Result Description	data: Personnel fingerprint data information; id: Personnel fingerprint ID; pin: Personnel number; validType: Validity status; bioType: Biometric template type; version: Fingerprint template version; currently available versions are 10.0 and 12.0; template: Fingerprint template content; templateNo: Fingerprint template number; templateNoIndex: Index corresponding to the fingerprint template.

2.26.3 Delete BioTemplate by Personnel ID

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

Response Result Description	Refer to General Response Result Parameter Description Interface Parameter
------------------------------------	--

2.26.4 Delete Personnel BioTemplate Based on Personnel ID and BioTemplate ID

Post Request URL	http://serverIP:serverPort/api/person/delete/{pin}{templateNo}?access_token={apitoken}
Request Mode	POST
Request Content	none
Request Parameter Description	pin: personnel ID, required; templateNo: BioTemplate ID, required
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.26.5 Retrieve BioTemplate Information by Personnel ID--V2

Post Request URL	http://serverIP:serverPort/api/v2/bioTemplate/getFgListByPin?access_token={apitoken}
Request Mode	GET
Request Content	none
Request Parameter Description	pin: personnel ID, required;
Response Result	{ "code": 0,

	<pre> "message": "success", "data": { "id": "40280bb470e118510170e13f57dc001a", "pin": "3170156", "validType": "1", "bioType": 1, "version": "10", "template": "TWITUzIxAAAkEikECAUHCc7QAAAcK2kBAAA AhNcowCrDALI P3.....(Biotemplate)", "templateNo": "6", "templateNoIndex": 0 } } </pre>
Response Result Description	data: Personnel fingerprint data information; id: Personnel fingerprint ID; pin: Personnel identification number; validType: Validity status; bioType: Biometric template type.

2.26.6 Delete BioTemplate Based on Personnel ID --V2

Post Request URL	http://serverIP:serverPort/api/v2/bioTemplate/deleteByPin?access_token={apitoken}&pin={pin}
Request Mode	POST
Request Content	none
Request Parameter	pin: personnel ID, required;

Description	
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.26.7 Delete Personnel BioTemplate Based on Personnel ID and BioTemplate ID --V2

Post Request URL	http://serverIP:serverPort/api/v2/person/delete?access_token={apitoken}&pin={pin}&templateNo={templateNo}
Request Mode	POST
Request Content	none
Request Parameter Description	pin: personnel ID, required; templateNo: BioTemplate ID, required
Response Result	Refer to the General Response Result Interface Parameter . For more details, see Appendix - Error Codes
Response Result Description	Refer to General Response Result Parameter Description Interface Parameter

2.27 Temperature Transactions Interface (Not Available in Some Versions)

2.27.1 Get Hep Transaction [api/hep/transaction/list]

Post Request URL	http://serverIP:serverPort/api/hep/transaction/list?personPin={personPin}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET

Request Content	None
Request Parameter Description	personPin: Pin (Not required) startDate: StartTime (Not required, query by time, work with endtime) endDate: EndTime (Not required, query by time, work with starttime) PageNo: page no, required, the value must be greater than 0; pageSize: the number of each page, required, the value must be greater than 0, not more than 1000 per page;
Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "id": "402855a6723b91d101723b9e3edf0004", "eventTime": "2020-05-22 17:01:50", "pin": "0", "name": "UnregisterUser", "lastName": "", "deptName": "NULL", "areaName": "Area Name", "cardNo": null, "devSn": "CKJF201260077", "verifyModeName": null, "eventName": "acc_newEventNo_68", "eventPointName": "", "readerName": null, "accZone": null, "devName": "192.168.213.160", "logId": null, "eventNumber": "511", "photoPath": "/upload/event/photo/CKJF201260077/20200522170150.jpg", "maskFlag": "1", }] }</pre>

	<pre> "temperature": 36.56, "status": "Normal" }, } </pre>
Response Result Description	data: transaction array list; id: "402856aa6c3c5063016c3cb5cfa2001d";record id 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; eventName: transaction description; eventPointName: transaction point name; readerName: reader name; accZone: access area name; devName: device name; logId: transaction index value, there will be a value if device support it; eventNumber: event record number photoPath: Instant capture photo storage path maskFlag: weather wear a mask (0: unmask, 1: wear mask, Null: Undetected) temperature: temperature status: body temperature status (normal, abnormal, unmeasured)

2.27.2 Get Mask and Temperature Real-Time Monitor Events

[api//hep/transaction/monitor]

Post Request URL	http://serverIP:serverPort/api/hep/transaction/monitor?timestamp={timestamp}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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
Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "id": "", "eventTime": "2020-07-03 16:57:16", "pin": "1", "name": "seven", "lastName": "wu", "deptName": "Department Name", "areaName": "Area Name", "cardNo": null, "devSn": "CKJF201260077", "eventName": "Normal Verify Open", "eventPointName": "192.168.218.202-1", "devName": "192.168.218.202", "eventNumber": "315", "photoPath": "/upload/event/photo/CKJF201260077/20200703165716-1.jpg", "maskFlag": "1", "temperature": 36.6,</pre>

	<pre>"status": "Normal" }} }</pre>
Response Result Description	data: transaction array list; id: "empty";There has empty event id in real time monitor. eventTime: Record device trigger time; pin: pin; name: name; lastName: person family name (English); deptName: department name; areaName: area name; cardNo: card no; devSn: device sn; eventName: transaction description; eventPointName: transaction point name; devName: Device name; eventNumber: event record number; photoPath: Instant capture photo storage path; maskFlag: weather wear a mask (0: unmask, 1: wear mask, Null: Undetected) temperature: temperature status: body temperature status (normal, abnormal, unmeasured)

2.28 FaceKiosk Personal Information

2.28.1 Edit Personal Information

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

Request Content	<pre>{ "category": 0, "perDevAuth": 0, "personPin": 1 }</pre>
Request Parameter Description	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
Response Result	<pre>{ "code": 0, "message": "success", "data": null }</pre>

2.29 FaceKiosk Area Personal

2.29.1 Delete Area Personnel

Post Request URL	<pre>http://serverIP:serverPort/api/insAreaPerson/delAreaPerson?access_token={apitoken}</pre>
Request Mode	POST
Request Content	<pre>{ "code": 1, "pins": "1,2,3" }</pre>
Request Parameter Description	code(string): Area code (For details/api/insAreaPerson/getAreaList) pins (string): Personnel number collection(Multiple personnel

	are separated by ',')
Response Result	<pre>{ "code":0, "message":"success", "data": null }</pre>
Response Result Description	

2.29.2 Add Area Personnel

Post Request URL	http://serverIP:serverPort/api/insAreaPerson/setAreaPerson?access_token={apitoken}
Request Mode	POST
Request Content	<pre>{ "code": 1, "pins": "1,2,3" }</pre>
Request Parameter Description	code (string): Area code (For details/api/insAreaPerson/getAreaList) pins (string): personnel number collection(Multiple personnel are separated by ',')
Response Result	<pre>{ "code":0, "message":"success", "data": null }</pre>

	<pre>"data": null }</pre>
Response Result Description	

2.29.3 Get a List of Attendance Zones

Post Request URL	http://serverIP:serverPort/api/insAreaPerson/getAreaList?pageNo=1&pageSize=10&access_token={apitoken}
Request Mode	GET
Request Content	
Request Parameter Description	
Response Result	<pre>{ "code":0, "message":"success", "data": [{ "id": "402855b27d4b9eb4017d4ba051690002", "code": "1", "name": "Area name" }] }</pre>
Response Result Description	id: Area ID code: Area code name: Area name

2.29.4 Get a list of attendance zones (With total number)

Post Request URL	http://serverIP:serverPort/api/v2/insAreaPerson/getAreaList?pageNo=1&pageSize=10&access_token={apitoken}
Request Mode	GET
Request Content	
Request Parameter Description	
Response Result	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 1, "total": 2, "data": [{ "id": "402855b27d4b9eb4017d4ba051690002", "code": "1", "name": "Area name" }], "offset": 0, "lastPage": false } }</pre>

Response Result Description	id: Area ID
	code: Area code
	name: Area name

2.30 FaceKiosk Device Interface

2.30.1 Get Facekiosk Device List [api/device/attAdDeviceList]

Post Request URL	http://serverIP:serverPort/api/device/attAdDeviceList?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pageNo: pageno, required, the value must be greater than 0; pageSize: the number of each page, required, must be greater than 0
Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "id": "402848327562dedd01756d555c726227", "sn": "6271203100008", "name": "FaceKiosk-1", "type": "Robot-Pad", "status": "1", "module": "ins" }] }</pre>

	<pre> "name": "174559110004", "type": "ZK-S1050", "status": "1", "module": "ins" }] } </pre>
Response Result Description	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;

2.30.2 Get Facekiosk Device List (With Total Number)

[api/device/attAdDeviceList]

Post Request URL	http://serverIP:serverPort/api/v2/device/attAdDeviceList?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pageNo: pageno, required, the value must be greater than0; pageSize: the number of each page, required, must be greater than 0
Response Result	<pre> { "code": 0, "message": "success", "data": { "page": 0, </pre>

	<pre> "size": 1, "total": 2, "data": [{ "id": "402848327562dedd01756d555c726227", "sn": "6271203100008", "name": "FaceKiosk-1", "type": "Robot-Pad", "status": "1", "module": "ins" }], "offset": 0, "lastPage": false } } </pre>
Response Result Description	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;

2.31 Entrance Control Device Interface

2.31.1 Get Entrance Control Device List [api/psgdDvice/list]

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

	Type: device type name; gateType: Entrance Control device type, 0-unknow(device is not connected to entrance control terminals); 1-Tripod Turnstile; 2-Flap Barrier; 3-Swing Barrier
--	---

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

Post Request URL	http://serverIP:serverPort/api/psgDevice/getBySn?sn={sn}&access_to ken={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	Sn: device sn, required;
Response Result	<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>
Response Result Description	Data: device object; Id: device id; Sn: device sn; Name: device name; 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
--	---

2.31.3 Get Entrance Control Device List (With Total Number)

[api/v2/psgDevice/list]

Post Request URL	http://serverIP:serverPort/api/v2/psgDevice/list?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pageNo: pageno, required, the value must be greater than0; pageSize: the number of each page, required, must be greater than 0
Response Result	<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, "lastPage": true } }</pre>

	<pre> } } </pre>
Response Result Description	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: device array list; Id: device id; Sn: device sn; Name: device name; state: device state, 1 enabled, 0 disabled; Module: module, access, attendance, elevator; Type: device type name; gateType: Entrance Control device type, 0-unknow(device is not connected to entrance control terminals); 1-Tripod Turnstile; 2-Flap Barrier; 3-Swing Barrier

2.32 Entrance Control Gate Interface

2.32.1 Get the State of All Gates [psgGate/allGateState]

Post Request URL	http://serverIP:serverPort/api/psgGate/allGateState?access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	None
Response Result	<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>
Response Result Description	Data: entrance control gate state array List; Id: gate id; Name: entrance control device name; DeviceId: device id; Connection: connection state, 0 offline, 1online, 2 disabled; gateState: gate state (0 offline-device is not connect to entrance control gate, default: 0, 1 Online, 2 Disabled, 3 Lockdown) 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) relay: Relay state (0-off, 1-Exit open,2-Entry Open, None-unknown)

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

Post Request URL	http://serverIP:serverPort/api/psgGate/gateStateById?gateId={gateId}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	gateId: Entrance Control gate id, example: "8a888e8f820060eb018200e488650abb"
Response Result	<pre> { "code": 0, </pre>

	<pre> "message": "success", "data": [{ "id": "8a888e8f820060eb018200e488650abb", "name": "10.8.14.234-1", "deviceId": "8a888e8f820060eb018200e487a50a67", "connect": "1", "gateState": "0", "alarm": "0", "relay": "0" }] </pre>
Response Result Description	Data: entrance control gate state array List; Id: gate id; Name: entrance control device name; Deviceld: device id; Connection: connection state, 0 offline, 1online, 2 disabled; gateState: gate state (0 offline-device is not connect to entrance control gate, default: 0, 1 Online, 2 Disabled, 3 Lockdown) 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) relay: Relay state (0-off, 1-Exit open,2-Entry Open, None-unknown)

2.32.3 Get Entrance Control Gate State by the SN of Device

[psgGate/gateStateBySn]

Post Request URL	http://serverIP:serverPort/api/psgGate/gateStateBySn?deviceSn={sn}&access_token={apitoken}
Request Mode	GET

Request Content	None
Request Parameter Description	Sn: device sn
Response Result	<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>
Response Result Description	Data: entrance control gate state array List; Id: gate id; Name: entrance control device name; DeviceId: device id; Connection: connection state, 0 offline, 1online, 2 disabled; gateState: gate state (0 offline-device is not connect to entrance control gate, default: 0, 1 Online, 2 Disabled, 3 Lockdown) 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) relay: Relay state (0-off, 1-Exit open,2-Entry Open, None-unknown)

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

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

2.32.5 Get Entrance Control Gate List [api/psgGate/list]

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

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

2.32.6 Get Entrance Control Gate List (With Total Number)

[api/v2/psgGate/list]

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

	<pre> "size": 2, "total": 1, "data": [{ "id": "8a888e238488391e018488703f0a0066", "name": "10.8.14.249-1", "deviceId": "8a888e238488391e018488703e4b0012" }], "offset": 0, "lastPage": true } </pre>
Response Result Description	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: entrance control gate array list; id: gate id; name: gate name; deviceId: device id;

2.32.7 Remote Open by Gate Id [psgGate/remoteOpenById]

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

Description

2.32.8 Remote Open by Gate Name [psgGate/remoteOpenByName]

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

2.33 Entrance Control Gate Interface

2.33.1 Delete Person Level [psgLevel/deleteLevel]

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

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

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

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

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

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

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

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

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

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

2.33.6 Get Entrance Control Access Level [/psgLevel/list]

Post Request URL	http://serverIP:serverPort/api/psgLevel/list?pageNo={pageNo}&pageSize={pageSize}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pageNo: page no, required, the value must be greater than 0; pageSize, the number of each page, required, must be greater than 0

Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "id": "402856aa6c3bbde1016c3bc2329200ff", "name": "public level group" }] }</pre>
Response Result Description	Data: level group array list; Id: level group id, string; such as "402856aa6c3bbde1016c3bc2329200ff" Name: level group name;

2.33.7 Sync Level Info to Device [psgLevel/syncLevel]

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

2.33.8 Sync Person Level [/psgLevel/syncPerson]

Post Request URL	http://serverIP:serverPort/api/psgLevel/syncPerson?pin={pin}&levelId
-------------------------	--

	ds={levelIds}&access_token={apitoken}
Request Mode	POST
Request Content	None
Request Parameter Description	Level Ids: level group id list, multiple comma separated, required, string; such as "402856aa6c3bbde1016c3bc2329200ff" Pin: pin, required
Response Result	Refer to public response result, other can refer to Appendix-Error code
Response Result Description	Public response parameter description

2.33.9 Get Entrance Control Access Level (With Total Number)

[api/v2/psgLevel/list]

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

	<pre>"name": "public level group" }], "offset": 0, "lastPage": true } }</pre>
Response Result Description	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: level group array list; Id: level group id, string; such as "402856aa6c3bbde1016c3bc2329200ff" Name: level group name;

2.33.10 Add Person Level By Level [api/v2/psgLevel/addLevelByLevel]

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

	<pre> "personnel ID pin1", "personnel ID pin2", "personnel ID pin3"] }] </pre>
Request Parameter Description	levelId: access level group id, required; Pins: personnel ID pin list, required;
Response Result	Refer to public Response Results
Response Result Description	Public response parameter description

2.33.11 Add Person Level By Person [api/v2/psgLevel/addLevelByPerson]

Post Request URL	http://serverIP:serverPort/api/v2/psgLevel/addLevelByPerson?access_token={apitoken}
Request Mode	POST
Request Content	<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>

	<pre>] }] </pre>
Request Parameter Description	Pin: personnel ID pin list, required; levelIds: access level group id, required;
Response Result	Refer to public Response Results
Response Result Description	Public response parameter description

2.34 Entrance Control Gate Reader Interface

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

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

	<pre>{ "id": "8a888e8f820060eb018200e488660abd", "gateId": "8a888e8f820060eb018200e488650abb", "name": "10.8.14.234-1-出", "readerNo": 2, "readerState": 1 }</pre>
Response Result Description	Data: reader array List; Id: reader id; gateId: entrance control gate id; Name: reader name; ReaderNo: readerNo; ReaderState: reader state, 0: in, 1: out;

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

[api/v2/psgReader/list]

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

	<pre> "total": 2, "data": [{ "id": "8a888e238488391e018488703f0b0067", "gateId": "8a888e238488391e018488703f0a0066", "name": "10.8.14.249-1-入", "readerNo": 1, "readerState": 0 }, { "id": "8a888e238488391e018488703f0b0068", "gateId": "8a888e238488391e018488703f0a0066", "name": "10.8.14.249-1-出", "readerNo": 2, "readerState": 1 }], "offset": 0, "lastPage": true } </pre>
Response Result Description	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;

2.34.3 Get Entrance Control Gate Reader Info by Reader Id

[psgReader/getByld]

Post Request URL	http://serverIP:serverPort/api/psgReader/getByld?id={id}&access_to ken={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	id: reader id, required;
Response Result	<pre>{ "code": 0, "message": "success", "data": { "id": "402856aa6c3c5063016c3cb3a0360005", "gateId": "402856aa6bff4b1a016bff5374800c0e", "name": "192.168.214.15-1-in", "readerNo": 1, "readerState": 0 } }</pre>
Response Result Description	data: reader info object; id: reader id; Gate Id: gate id; name: reader name; readerNo: reader no; readState: reader state, 0: in, 1:out;

2.35 Entrance Control Device transaction Interface

2.35.1 Get Entrance Control Transaction by Device Sn

[api/psgTransaction/device/{deviceSn}]

Post Request URL	http://serverIP:serverPort/api/psgTransaction/device/{deviceSn}?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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; startDate: start time, not required; endDate: end time, not required;
Response Result	<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", }] }</pre>

	<pre> "eventPointName": "10.8.14.234-1", "readerName": "Others", "devName": "10.8.14.234", "logId": 3603, "eventNo": 110, "gateType": 0 }] } </pre>
Response Result Description	data: transaction array list; id: event id, "402856aa6c032ca3016c03351d2600fd"; eventTime: record device trigger time; pin: pin; name: person name; lastName: person last name, (English); 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

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

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

Post Request URL	http://serverIP:serverPort/api/v2/psgTransaction/device/{deviceSn}?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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; startDate: stat time, not required; endDate: end time, not required;
Response Result	<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>
Response Result Description	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;

	name: person name; lastName: person last name, (English); 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
--	--

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

[api/v3/psgTransaction/device]

Post Request URL	http://serverIP:serverPort/api/v3/psgTransaction/device/{deviceSn}?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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; startDate: stat time, not required; endDate: end time, not required;
Response Result	{ "code": 0,

```
"message": "success",

"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,
```

	<pre> "gateType": 0 }, "offset": 0, "lastPage": true } }</pre>
Response Result Description	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; name: person name; lastName: person last name, (English); 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

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

Post Request URL	http://serverIP:serverPort/api/psgTransaction/list?personPin={personPin}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	personPin: personPin, [not required] startDate: startDate, [not required] endDate: endDate, [not required] 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;
Response Result	<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", "readerName": "Others", "devName": "10.8.14.234", "logId": 3603, }] }</pre>

	<pre> "eventNo": 110, "gateType": 0 }] }</pre>
Response Result Description	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; 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

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

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

Request Content	None
Request Parameter Description	personPin: personPin, [not required] startDate: startDate, [not required] endDate: endDate, [not required] 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;
Response Result	<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 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 }], } }</pre>

	<pre> "offset": 0, "lastPage": true } } </pre>
Response Result Description	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; 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

2.35.6 Get Entrance Control Device Real-Time Monitoring Transaction

[api/psgTransaction/monitor]

Post Request URL	http://serverIP:serverPort/api/psgTransaction/monitor?timestamp={timestamp}&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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
Response Result	<pre>{ "code": 0, "message": "success", "data": [{ "id": null, "eventTime": "2022-07-18 18:13:33", "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, </pre>

	<pre> "gateType": 0 }] } </pre>
Response Result Description	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

2.35.7 Get Personnel Entrance Control Access Transaction by Pin

[api/psgTransaction/person/{pin}]

Post Request URL	http://serverIP:serverPort/api/psgTransaction/person/{pin}?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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]
Response Result	<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", "readerName": "Others", "devName": "10.8.14.234", "logId": 3603, "eventNo": 110, "gateType": 0 }] } </pre>
Response Result Description	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
--	---

2.35.8 Get Entrance Control Gate Transaction List (With Total Number)

[api/v2/psgTransaction/list]

Post Request URL	http://serverIP:serverPort/api/v2/psgTransaction/list?personPin={personPin}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	personPin: personPin, [not required] startDate: startDate, [not required] endDate: endDate, [not required] 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;
Response Result	<pre>{ "code": 0, "message": "success", "data": { "page": 1, "size": 1, </pre>

	<pre> "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 }], "offset": 0, "lastPage": true } </pre>
Response Result Description	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
--	--

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

Post Request URL	http://serverIP:serverPort/api/v3/psgTransaction/person/{pin}?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	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]
Response Result	<pre>{ "code": 0, "message": "success", "data": {</pre>

	<pre> "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 }], "offset": 0, "lastPage": true } </pre>
Response Result Description	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 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: 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
--	---

2.36 Parking Device Interface

2.36.1 Retrieve the Device Channel List

Post Request URL	http://serverIP:serverPort/api/v1/parkBase/getParkChannelList?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request Content	None
Request Parameter Description	pageNo: Page, required, Must be greater than 0; pageSize: page size, required, Must be greater than 0;
Response Result	<pre>{ "code": 0, "message": "success", "data": { "data": [{</pre>

	<pre> "channelName": "Exit", "channelId": "4028b59c82101c36018210a7aabb05a0" }, { "channelName": "Entrance", "channelId": "4028b59c81faa6120181fb2207e803bc" }], "totalCount": 2 } } </pre>
Response Result Description	data: Device array collection; id: Device channel ID; name: Device channel name.

2.36.2 The Specified Device Channel is Opened

Post Request URL	http://serverIP:serverPort/api/v1/parkBase/openGateChannel?channelId=4028b59c81faa6120181fb2207e803bc&access_token={apitoken}
Request Mode	POST
Request Content	not have
Request Parameter Description	channelId: Channel id, required; e. g., "4028b59c81faa6120181fb2207e803bc"
Response Result	See the Universal Response Results
Response Result Description	Description of the generic response parameters

2.36.3 New and Edit Parking Personnel License Plates

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

2.36.4 Delete the Parking Persons License Plates

Post Request URL	http://serverIP:serverPort/api/v1/parkPlate/delPlate?access_token={apitoken}
Request Mode	POST
Request Content	<pre>{ "pin": "1234", "carNumber": "Min A32212" }</pre>

	}
Request Parameter Description	carNumber: New license plate number, such as: Min A32212 "pin": personnel pin, such as: 12345
Response Result	See general Response Results and others see Appendix-Exception Dictionary
Response Result Description	Description of the generic response parameters

2.37 Offline Consumption and Consumption Record Interface

2.37.1 Obtain the Offline Consumption Records

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

```
"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,
```

	<pre> " subBalance": "9.98", " posMode": 1, "hallingCode": "1", "hallingName": "Headquarters", "mealCode": "1", "mealName": "Breakfast", "devSn": "10001", "devAlias": "canteen consumer machine", " devSerialNum": 1088 }], "offset": 0, "lastPage": true } } </pre>
Response Result Description	<pre> posTime Consumption time personPin Personnel number personName Name of the person personLastName Persons last name sysCardNo Card account number cardStatus Card status (1 valid 3 loss reporting 4 consumption expired 5 shutdown) cardSerialNum Card serial number deptCode Department number </pre>

deptName	Department name
Card	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

2.38 Online Consumption Account Interface

2.38.1 Get the Personnel Account Information through the Personnel Number

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

Request Mode	GET
Request content	not have
Request Parameter Description	personPin: Personnel number;
Response Results	<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>
Response Result Description	personPin Personnel number personName Name of the person personLastName Persons last name accountNo Account number balance Balance (points) posMoneyWallet Cash Wallet balance (points) posAllowanceWallet Subsidy wallet balance (points) deptCode Department number deptName Department name consumeSerailnum Account serial number available Is valid 0 enabled 1 disabled accountMember Is it a member account 0 non-member 1 member accountKindName Account category name

2.39 Online Consumption and Consumption Record Interface

2.39.1 Get Consumption Records

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

Request content	not have
Request Parameter Description	personPin: Personnel number (not required); startDate: Start time of the query record (not required); endDate: End time of query record (not required); pageNo: Page number, required, must be greater than 0; pageSize The number of articles on each page, required, must be greater than 0;
Response Results	<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>
Response Result Description	<pre> personPin Personnel number personName Name of the person personLastName Persons last name accountNo Account number typeName Type name (6. Consumption 8. replenishment 9. error correction 10. count 11. bookkeeping 12 offline consumption 13 cancel ordering) money Consumption amount (points) balance Balance (points) moneyWallet Cash Wallet balance (points) allowanceWallet Subsidy wallet balance (points) 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) discount Discount discountMoney Discount amount consumeOrder Consumption order deptCode Department number deptName Department name hallingCode Restaurant number hallingName Restaurant name </pre>

	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)

2.40 Smart Video Device Interface

2.40.1 Obtain the Main Device Interface

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

	<pre> "message": "success", "data": { "page": 0, "size": 20, "total": 1, "data": [{ "id": "8a888ee28dba4ebc018dba51d1180001", "ip": "10.8.14.126", "status": "1", "userName": "admin", "passWord": "eav1a\$calRqfngfSSP5RIQqTjlsA==", "port": "8081", "type": "7" }], "offset": 0, "lastPage": true } } </pre>
Response Result Description	Id: primary device id Ip: primary device ip status: Online status

	userName: Account name
	passWord: password
	port: port
	Type: Primary device type

2.40.2 Get the Device Interface

Post Request URL	http://serverIP:serverPort/api/ivsDevice/getDevList?pageNo=1&pageSize=20&parentId=[parentId]&access_token={apitoken}
Request Mode	GET
Request content	not have
Request Parameter Description	pageNo: Page number, required, must be greater than 0; pageSize: Number of articles on each page, required, must be greater than 0; parentId: Main device id, required
Response Results	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 1, "total": 18, "data": [{</pre>

	<pre> "id": "8a888ee28dba4ebc018dba51d67a0003", "cameraName": "CH1", "cameraType": "DOME", "status": "0", "channel": "1", "parentType": "7", "channelCode": "CH1", "vidParentDeviceId": "8a888ee28dba4ebc018dba51d1180001", "channelId": "8a888ee28dba4ebc018dba524f6f0015" }], "offset": 0, "lastPage": false } }</pre>
Response Result Description	Id: equipment id cameraName: Device name cameraType: Device type status: Online status channel: Channel number parentType: Main device type channelCode: Channel number vidParentDeviceId: Main device id

	channelId: channel id
--	-----------------------

2.40.3 Gets the Device Live Stream Address

Post Request URL	http://serverIP:serverPort/api/ivsDevice/getDeviceRtsp?channelId=[channelId]&rtspType=[rtspType]&access_token={apitoken}
Request Mode	GET
Request content	not have
Request Parameter Description	channelId: Channel id, required; rtspType: Code stream type, required, 0: all code stream, 1: main code stream, 2: auxiliary code stream;
Response Results	<pre>{ "code": 0, "message": "success", "data": { "resultCode": "0", "content": { "rtspUrls": [{ "rtspUrl": "rtsp://admin:Admin123@10.8.14.126:554/ch02.264?ptype=udp", "type": 1, "parentType": 7 }] } } }</pre>

	<pre> "rtspUrl": "rtsp://admin:Admin123@10.8.14.126:554/ch02_sub.264?ptype=udp", "type": 2, "parentType": 7 }] } } } </pre>
Response Result Description	rtspUrls: Code stream list rtspUrl: Code stream address Type: code stream type parentType: Main device type

2.40.4 Gets the Device Playback List

Post Request URL	http://serverIP:serverPort/api/ivsDevice/getVideoList?channelId=[channelId]&startTime=[startTime]&endTime=[endTime]&access_token={apitoken}
Request Mode	GET
Request content	not have
Request Parameter Description	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;
Response Results	{

	<pre> "code": 0, "message": "success", "data": { "videoItemList": [{ "startTime": "2024-02-19 05:41:02", "endTime": "2024-02-19 07:44:32", "type": "PLAN" }, { "startTime": "2024-02-19 07:45:40", "endTime": "2024-02-19 16:37:40", "type": "PLAN" }] } </pre>
Response Result Description	videoItemList: Playback list startTime: Start time endTime: End time Type: video type, PLAN: planned video, ALARM: alarm video

2.40.5 Obtain the Device Back to the Release Address

Post Request URL	http://serverIP:serverPort/api/ivsDevice/getRtspPlayBack?channelId=[channelId]&startTime=[startTime]&endTime=[endTime]&access_token={apitoken}
Request Mode	GET
Request content	not have
Request Parameter Description	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; (the time interval is in the list obtained in the previous interface, no video can not be broadcast)
Response Results	<pre>{ "code": 0, "message": "success", "data": { "passWord": "Admin123", "rtspUrl": "rtsp://admin:Admin123@10.8.14.126:554/recording?ch=2&stream=1&start=20240219074432&stop=20240219054102", "vidParentDeviceId": "8a888ee28dba4ebc018dba51d1180001", "rtspUrls": [{ "rtspUrl": "rtsp://admin:Admin123@10.8.14.126:554/recording?ch=2&stream=0&start=20240219074432&stop=20240219054102",</pre>

	<pre> "type": 1 }, { "rtspUrl": "rtsp://admin:Admin123@10.8.14.126:554/recording?ch=2&stream=1&start=202402 19074432&stop=20240219054102", "type": 2 }], "type": "7", "userName": "admin", "channelCode": "CH2" } } </pre>
Response Result Description	rtspUrls: Code stream list rtspUrl: Code stream address Type: code stream type vidParentDeviceId: Main device id userName: Account name passWord: password channelCode: Channel number Type: Primary device type

2.40.6 Set the Push Address

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

2.41 Parking Basic Management Interface

2.41.1 List of Parking Yard Areas (v2)

Post Request URL	http://serverIP:serverPort/api/v2/parkBase/getParkingAreaList?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET

Request content	NONE
Request Parameter Description	pageNo: Page number, required, must be greater than 0; pageSize The number of articles on each page, required, must be greater than 0;
Response Results	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 10, "total": 1, "data": [{ "id": "4028b59c8d3a34be018d3a5d1b2e0009", "name": "B1", "parentId": "" }], "offset": 0, "lastPage": true } }</pre>
Response	Data: array sets;

Result Description	id: parking area id; parentId: parent parking area id; name: parking area name;
---------------------------	---

2.41.2 Get Parking Discount Strategy List (v2)

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

	<pre> "discountType": 3, "discountId": "4028b59c919c3c4001919ccdf898000c", "content": "120" }, { "discountName": "Amount", "discountType": 2, "discountId": "4028b59c919c3c4001919ccda746000a", "content": "10.0" }, { "discountName": "Free at all", "discountType": 1, "discountId": "4028b59c919c3c4001919cccfa010007", "content": "0" }], "offset": 0, "lastPage": true } } </pre>
Response Result Description	Data: array sets; discountId: Discount id; discountType: 1, discount type

	discountName: Name; Content: Discount content
--	--

2.41.3 Parking Entry and Exit List (v2)

Post Request URL	http://serverIP:serverPort/api/v2/parkBase/getParkingEntranceList?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request content	not have
Request Parameter Description	pageNo: Page number, required, must be greater than 0; pageSize The number of articles on each page, required, must be greater than 0;
Response Results	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 10, "total": 1, "data": [{ "id": "4028b59c89d90a5e0189d987deac002f", "name": "A-E1", "areald": "4028b59c89bf49dd0189bf50ceb7004e",</pre>

	<pre> }], "offset": 0, "lastPage": true } }</pre>
Response Result Description	Data: array sets; id: Regional import and export id; name: name; areald: region id;

2.41.4 List of Parking Vehicle Types (v2)

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

	<pre> "page": 0, "size": 10, "total": 1, "data": [{ "id": "4028b59c8d3a34be018d3a5d1b2e0009", "Name": "Small car A" }], "offset": 0, "lastPage": true } } </pre>
Response Result Description	Data: array sets; id: type id; name: name;

2.41.5 Parking Space List (v2)

Post Request URL	http://serverIP:serverPort/api/v2/parkBase/getParkingSpaceList?pageNo=1&pageSize=20&areald=4028b59a788258f501788262e5d209ed&access_token={apitoken}
Request Mode	GET
Request content	None

Request Parameter Description	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;
Response Results	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 10, "total": 1, "data": [{ "id": "4028b59c919cd64201919cddab770106", "name": "K005", "areald": "4028b59a788258f501788262e5d209ed" }], "offset": 0, "lastPage": true } }</pre>
Response Result	Data: array sets; id: parking space id;

Description	Name: Parking Space name; areald: parking area id;
--------------------	---

2.41.6 List of Parking Channels (v2)

Post Request URL	http://serverIP:serverPort/api/v2/parkBase/getParkingChannelList?pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request content	None
Request Parameter Description	pageNo: Page number, required, must be greater than 0; pageSize The number of articles on each page, required, must be greater than 0;
Response Results	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 10, "total": 1, "data": [{ "channelName": "Export", "channelId": "4028b59c82101c36018210a7aabb05a0" }] } }</pre>

	<pre>], "offset": 0, "lastPage": true } } </pre>
Response Result Description	Data: array sets; channelId: parking channel id; channelName: Parking channel name;

2.41.7 Vehicle Admission List (v2)

Post Request URL	http://serverIP:serverPort/api/v2/parkTransaction/listParkRecordin?carNumber={carNumber}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request content	not have
Request Parameter Description	carNumber: License plate number, not required startDate: Start time, not required, together with the end time ("yyyy-MM-dd HH:mm:ss") endDate: End time, not required, together with the start time ("yyyy-MM-dd HH:mm:ss") 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;
Response	{

Results

```
"code": 0,

"message": "success",

"data": {

  "page": 0,

  "size": 1,

  "total": 1093,

  "data": [

    {

      "id": "4028b59c919255c40191928f905000f4",

      "parkingAreaName": "Development Zone",

      "name": "",

      "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

}
```

Response Result Description	id: "402856aa6c1778b3016c1a695c860018", parkingAreaName: Parking area; Name: Name of the car owner,; carNumber: License plate number; 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) checkInTime: Admission time
------------------------------------	---

2.41.8 Vehicle Exit List (v2)

Post Request URL	http://serverIP:serverPort/api/v2/parkTransaction/listParkRecordout?carNumber={carNumber}&startDate={startDate}&endDate={endDate}&pageNo=1&pageSize=20&access_token={apitoken}
Request Mode	GET
Request content	not have
Request Parameter Description	carNumber: License plate number, not required startDate: Start time, not required, together with the end time ("yyyy-MM-dd HH:mm:ss") endDate: End time, not required, together with the start time ("yyyy-MM-dd HH:mm:ss")

	mm: ss") pageNo: Page number, required, must be greater than 0; pageSize: The number of each page, must be greater than 0, each page shall not exceed 1000;
Response Results	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 1, "total": 1856, "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": "-" }] } }</pre>

	<pre> }], "offset": 0, "lastPage": false } } </pre>
Response Result Description	Data: array sets; id: "402856aa6c1e1def016c1e2073e80001"; parkingAreaName: Parking area; Name: Name of the car owner,; carNumber: License plate number; 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

2.41.9 Edit the Parking Allowed / Prohibited List (v2)

Post Request URL	http://serverIP:serverPort/api/v2/parkPlate/addAllowDisabled?type={ type}&carNumber={ carNumber}&oldCarNumber={oldCarNumber}&parkCode={parkCode}&fromTi
-----------------------------	--

	me={ fromTime}&toTime={ toTime}&access_token={apitoken}
Request Mode	POST
Request content	not have
Request Parameter Description	Type: allowed / prohibited list type, 0: allowed list, 1: prohibited list carNumber: New license plate number, such as: Min A32212 oldCarNumber: Old license plate number, such as: not required parkCode: Car id, such as: not required fromTime: Start time, 2018-08-17 toTime: End time 2019-08-16
Response Results	See General Response Results and others see Appendix-Exception Dictionary
Response Result Description	Description of the generic response parameters

2.41.10 Parking of Authorized Vehicle List (v2)

Post Request URL	http://serverIP:serverPort/api/v2/parkPlate/addFixedVehicle?access_token={apitoken}
Request Mode	POST
Request content	{ "amount": "0", "areald": "Area Id", "carNumber": "Liao A12345",

	<pre> "carTypeId": "CarType Id", "endTime": "YYYY-MM-DD", "entranceIds": "1,2,3", "pin": "Person Pin", "spaceId": "Space Id", "startTime": "YYYY-MM-DD" } </pre>
Request Parameter Description	<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>
Response Results	<pre> { "code": 0, "message": "success", "data": null } </pre>
Response Result	Description of the generic response parameters

Description	
--------------------	--

2.41.11 Parking Delete Authorized Vehicle List (v2)

Post Request URL	http://serverIP:serverPort/api/v2/parkPlate/delFixedVehicle?access_token={apitoken}
Request Mode	POST
Request content	{ "carNumber": "Liao A12345", "pin": "Person Pin" }
Request Parameter Description	"carNumber": "A12345", "pin": "Person Pin"
Response Results	{ "code": 0, "message": "success", "data": null }
Response Result Description	Description of the generic response parameters

2.42 Space Device Interface

2.42.1 Get Space Device Info by Sn

Post Request URL	http://serverIP:serverPort/api/spaceDevice/getBySn?sn={sn}&access_token={apitoken}
Request Mode	GET
Request content	None
Request Parameter Description	Sn: device serial number, required;
Response Results	<pre>{ "code": 0, "message": "success", "data": { "id": "8a8890e192d0aee0192dc804b356b92", "sn": "500152130033", "name": "10.8.16.251", "type": "FaceKiosk-H10A", "status": "0", "module": "space" } }</pre>
Response	Data: device information;

Result Description	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;
---------------------------	---

2.42.2 Get Device List

Post Request URL	http://serverIP:serverPort/api/spaceDevice/list?access_token={apitoken}&deviceName={deviceName}&sn={sn}&pageNo={pageNo}&pageSize={pageSize}
Request Mode	GET
Request content	None
Request Parameter Description	deviceName: device name; Sn: device serial number, required; 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;
Response Results	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 50, </pre>

	<pre> "total": 1, "data": [{ "id": "8a8890e192e039c60192e073913f4675", "sn": "500152110018", "name": "10.8.53.104", "type": "FaceKiosk-H10A", "status": "0", "module": "space" }], "offset": 0, "lastPage": true } } </pre>
Response Result Description	Data: array sets; Id: device id; Sn: device sn; Name: device name; Type: device model; Status: device status, 0: offline, 1: online, 2: disabled;

2.43 Space Interface

2.43.1 Get Space List

Post Request URL	http://serverIP:serverPort/api/space/list?access_token={apitoken}&name={name}&pageNo={pageNo}&pageSize={pageSize}
Request Mode	GET
Request content	None
Request Parameter Description	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;
Response Results	<pre>{ "code": 0, "message": "success", "data": { "page": 0, "size": 50, "total": 1, "data": [{ "id": "8a8890e1936b8e2601936bb8d7380048", "name": "test space1", "picture": "" }] } }</pre>

	<pre> "status": "0", "capacity": "21", "spaceFacilityIds": "8a8890e192b76ffd0192b771806537a6,8a8890e192b76ffd0192b771806637aa", "spaceDeviceIds": "8a8890e192e039c60192e073913f4675", "authAreaName": "Area Name", "address": "D09 七楼 1011", "personId": "8a8890e1932832af019328c3ff422a94", "personPin": "3170079", "personTel": "151234567891", "smartSpace": true, "remark": "", "retentionTime": "15" }], "offset": 0, "lastPage": true } } </pre>
Response Result Description	Data: array sets; Id: space id; Name: space name; Picture: space photo base64 string

	Status: status, 0: space, 1: in using, 2: disabled;
	Capacity: capacity
	spaceFacilityIds: Facility ID, multiple IDs are separated by commas;
	spaceDeviceIds: deviceid, multiple IDs are separated by commas;
	authAreaName: area name;
	address: space address;
	personId: manager person id;
	personPin: manager person pin;
	personTel: manager person's contact information;
	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);
	remark: space description;
	retentionTime: retention duration, four options: 15, 30, 45, 60 minutes, if it is empty, no space retention duration is set;

2.43.2 Query Space Information Based on Space ID

Post Request URL	http://serverIP:serverPort/api/space/getById?id={id}&access_token={apitoken}
Request Mode	GET
Request content	None
Request Parameter Description	Id: space id, required;

Response Results	<pre>{ "code": 0, "message": "success", "data": { "id": "8a8890e1936b8e2601936bb8d7380048", "name": "test space1", "picture": "", "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>
Response Result	Data: space information;

Description	Id: space id; Name: space name; Picture: space photo base64 string Status: status, 0: space, 1: in using, 2: disabled; Capacity: capacity spaceFacilityIds: Facility ID, multiple IDs are separated by commas; spaceDeviceIds: deviceid, multiple IDs are separated by commas; authAreaName: area name; address: space address; personId: manager person id; personPin: manager person pin; personTel: manager person's contact information; 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); remark: space description; retentionTime: retention duration, four options: 15, 30, 45, 60 minutes, if it is empty, no space retention duration is set;
--------------------	--

2.43.3 Delete Space Based on Space ID

Post Request URL	http://serverIP:serverPort/api/space/deleteById?&access_token={apitoken}
Request Mode	POST
Request	{

content	<pre>"id": "8a8890e1936b8e2601936bb8d7380048" }</pre>
Request Parameter Description	Id: space id, required;
Response Results	<pre>{ "code": 0, "message": "Succeed: 1, Failed: 0.", "data": [] }</pre>
Response Result Description	Description of the generic response parameters

2.43.4 Add/Edit Space

Post Request URL	http://serverIP:serverPort/api/space/addSpace?access_token={apitoken}
Request Mode	POST
Request content	<pre>{ "address": "D09 7 floor 101", "capacity": 20, "id": "", "name": "test space", "personPin": 1234567, }</pre>

	<pre> "personTel": 15123456789, "picture": "", "remark": "this is a space", "retentionTime": 15, "smartSpace": false, "spaceDeviceIds": "8a8890e192e039c60192e073913f4675", "spaceFacilityIds": "8a8890e192b76ffd0192b771806637aa,8a8890e192b76ffd0192b771806537a6" } </pre>
Request Parameter Description	Id: space id, required, empty when adding, not empty when editing; Name: space name, required; Picture: space photo base64 string Status: status, 0: space, 1: in using, 2: disabled; Capacity: capacity for the space, required; spaceFacilityIds: Facility ID, multiple IDs are separated by commas; spaceDeviceIds: deviceid, multiple IDs are separated by commas, required; address: space address; personPin: manager person pin, required; personTel: manager person's contact information; 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); remark: space description; retentionTime: retention duration, four options: 15, 30, 45, 60 minutes, if it is empty,

	no space retention duration is set;
Response Results	<pre>{ "code": 0, "message": "success", "data": null }</pre>
Response Result Description	Description of the generic response parameters

2.44 Space Facility Interface

2.44.1 Get Space Facility List

Post Request URL	http://serverIP:serverPort/api/spaceFacility/list?access_token={apitoken}&name={name}&pageNo={pageNo}&pageSize={pageSize}
Request Mode	GET
Request content	None
Request Parameter Description	Name: space facility 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;
Response Results	{

```
"code": 0,

"message": "success",

"data": {

  "page": 0,

  "size": 50,

  "total": 3,

  "data": [

    {

      "id": "8a8890e19371ca56019371deb3850010",

      "name": "test facility1",

      "facilityNo": 22,

      "remark": "this is a facility11",

      "initFlag": false

    },

    {

      "id": "8a8890e192b76ffd0192b771806637aa",

      "name": "Projector",

      "facilityNo": 1,

      "remark": "",

      "initFlag": true

    },

    {

      "id": "8a8890e192b76ffd0192b771806537a6",
```

	<pre> "name": "Electronic Whiteboard", "facilityNo": 2, "remark": "", "initFlag": true }], "offset": 0, "lastPage": true } } </pre>
Response Result Description	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;

2.44.2 Query Facility Information Based on Facility ID

Post Request URL	http://serverIP:serverPort/api/spaceFacility/getByld?id={id}&access_token={apitoken}
Request Mode	GET
Request content	None

Request Parameter Description	Id: facility id, required;
Response Results	<pre>{ "code": 0, "message": "success", "data": { "id": "8a8890e19371ca56019371deb3850010", "name": "test facility1", "facilityNo": 22, "remark": "this is a facility11", "initFlag": false } }</pre>
Response Result Description	Data: facility information; Id: facility id; Name: facility name; facilityNo: facility number; Remark: facility description; initFlag: Whether to initialize data;

2.44.3 Delete Facility Based on Facility ID

Post Request URL	http://serverIP:serverPort/api/spaceFacility/deleteById?id={id}&access_token={apitoken}
-----------------------------	---

Request Mode	POST
Request content	{ "id": "8a8890e19371ca56019371d7d0020009" }
Request Parameter Description	Id: facility id, required;
Response Results	{ "code": 0, "message": "Succeed: 1, Failed: 0.", "data": [] }
Response Result Description	Description of the generic response parameters

2.44.4 Add/Edit Space Facility

Post Request URL	http://serverIP:serverPort/api/spaceFacility/addFacility?access_token={apitoken}
Request Mode	POST
Request content	{ "facilityNo": 20, "id": "", "name": "test facility", }

	<pre>"remark": "this is a facility" }</pre>
Request Parameter Description	Id: facility id, required, empty when adding, not empty when editing; Name: facility name, required; facilityNo: facility number, required, integer greater than 20; remark: facility description;
Response Results	<pre>{ "code": 0, "message": "success", "data": null }</pre>
Response Result Description	Description of the generic response parameters

2.45 Space Service Interface

2.45.1 Get Space Service List

Post Request URL	http://serverIP:serverPort/api/spaceService/list?access_token={apitoken}&name={name}&pageNo={pageNo}&pageSize={pageSize}
Request Mode	GET
Request content	None

Request Parameter Description	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;
Response Results	<pre>{ "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 }] } }</pre>

	<pre> }, { "id": "8a8890e192b76ffd0192b77180413754", "name": "Snack", "remark": "", "initFlag": true }], "offset": 0, "lastPage": true } } </pre>
Response Result Description	Data: array sets; Id: service id; Name: service name; remark: service description; initFlag: Whether to initialize data, true: initialize data, false: not initialize data;

2.45.2 Query Facility Information Based on Facility ID

Post Request URL	http://serverIP:serverPort/api/spaceService/getById?id={id}&access_token={apitoken}
Request Mode	GET
Request	None

content	
Request Parameter Description	Id: service id, required;
Response Results	<pre>{ "code": 0, "message": "success", "data": { "id": "8a8890e19370e2c70193715870d10026", "name": "test service ", "remark": "this is a service1", "initFlag": false } }</pre>
Response Result Description	Data: service information; Id: service id; Name: service name; Remark: service description; initFlag: Whether to initialize data;

2.45.3 Delete Service Based on Service ID

Post Request URL	http://serverIP:serverPort/api/spaceService/deleteById?id={id}&access_token={apitoken}
Request Mode	POST

Request content	<pre>{ "id": "8a8890e19370e2c70193715870d10026" }</pre>
Request Parameter Description	Id: service id, required;
Response Results	<pre>{ "code": 0, "message": "Succeed: 1, Failed: 0.", "data": [] }</pre>
Response Result Description	Description of the generic response parameters

2.45.4 Add/Edit Space Service

Post Request URL	http://serverIP:serverPort/api/spaceService/addService?access_token={apitoken}
Request Mode	POST
Request content	<pre>{ "id": "", "name": "test service", "remark": "this is a service" }</pre>

Request Parameter Description	Id: service id, required, empty when adding, not empty when editing; Name: service name, required; remark: facility description;
Response Results	{ "code": 0, "message": "success", "data": null }
Response Result Description	Description of the generic response parameters

2.46 Space Reservation Interface

2.46.1 Get Space Reservation List

Post Request URL	http://serverIP:serverPort /api/spaceReservation/list?access_token={apitoken}&spaceName={spaceName}&subject={subject}&startTime={startTime}&endTime={endTime}&pageNo={pageNo}&pageSize={pageSize}
Request Mode	GET
Request content	None
Request Parameter Description	spaceName: space name; Subject: subject name; startTime: time range query start time;

	endTime: Time range query end time; 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;
Response Results	<pre> { "code": 0, "message": "success", "data": { "page": 0, "size": 50, "total": 1, "data": [{ "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, </pre>

	<pre> "applyPersonId": "8a8890e1932832af019328c3ff422a94", "applyPersonPin": "317009", "applyPersonName": "xxxxx", "applyPersonLastName": "lee", "applyPersonEmail": "rwq@zzz", "serviceIds": "", "reservationStatus": 3, "remindTime": "5m" },], "offset": 0, "lastPage": true } } </pre>
Response Result Description	Data: array sets; subject: subject; content: reservation content; spaceId: space id; spaceName: space name; spaceAddress: space address; startSignInDateTime: start sign-in time; startTime: reservation start time; endTime: reservation end time;

	startSignInTime: how many minutes in advance to sign in, four options: 0, 5, 15, 30 (unit: minutes); applyPersonId: apply person id; applyPersonPin: apply person pin; applyPersonName: apply person name; applyPersonLastName: apply person's last name; applyPersonEmail: apply person's email; serviceIds: space service ID, multiple service IDs are separated by commas; reservationStatus: reservation status, 0: to be started, 1: in progress, 2: ended, 3: cancelled; 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;
--	--

2.46.2 Query Reservation Information Based on Reservation ID

Post Request URL	http://serverIP:serverPort/api/spaceReservation/getById?id={id}&access_token={apitoken}
Request Mode	GET
Request content	None
Request Parameter Description	Id: reservation id, required;
Response Results	<pre>{ "code": 0, "message": "success", }</pre>

```

"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",

    "applyPersonPin": "3170079",

    "applyPersonName": "jeiffu",

    "applyPersonLastName": "lee",

    "applyPersonEmail": "rwq@zzz",

    "participantIds": null,

    "serviceIds": "",

    "reservationStatus": 3,

    "remindTime": "5m"

}

```

Response

Data: reservation information;

Result Description	subject: subject; content: reservation content; spaceId: space id; spaceName: space name; spaceAddress: space address; startSignInDateTime: start sign-in time; startTime: reservation start time; endTime: reservation end time; startSignInTime: how many minutes in advance to sign in, four options: 0, 5, 15, 30 (unit: minutes); applyPersonId: apply person id; applyPersonPin: apply person pin; applyPersonName: apply person name; applyPersonLastName: apply person's last name; applyPersonEmail: apply person's email; serviceIds: space service ID, multiple service IDs are separated by commas; reservationStatus: reservation status, 0: to be started, 1: in progress, 2: ended, 3: cancelled; 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;
---------------------------	--

2.46.3 Delete Reservation Based on Reservation ID

Post Request URL	http://serverIP:serverPort/api/spaceReservation/deleteById?id={id}&access_token={apitoken}
Request Mode	POST

Request content	{ "id": "8a8890e19376a2ba019376a972860014" }
Request Parameter Description	Id: reservation id, required;
Response Results	{ "code": 0, "message": "Succeed: 1, Failed: 0.", "data": [] }
Response Result Description	Description of the generic response parameters

2.46.4 Cancel Reservation Based on Reservation ID

Post Request URL	http://serverIP:serverPort/api/spaceReservation/cancelReservation?access_token={apitoken}
Request Mode	POST
Request content	{ "id": "8a8890e19399fa3701939a02f33a0015" }
Request Parameter Description	Id: reservation id, required;

Response Results	<pre>{ "code": 0, "message": "Succeed: 1, Failed: 0.", "data": [] }</pre>
Response Result Description	Description of the generic response parameters

2.46.5 Finish Reservation Based on Reservation ID

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

Response Result Description	Description of the generic response parameters
------------------------------------	--

2.46.6 Query Reservation Participants Based on Reservation ID

Post Request URL	http://serverIP:serverPort/api/spaceReservation/getReservationParticipants?access_token={apitoken}
Request Mode	POST
Request content	<pre>{ "id": "8a8890e19399fa3701939a02f33a0015" }</pre>
Request Parameter Description	Id: reservation id, required;
Response Results	<pre>{ "code": 0, "message": "success", "data": [{ "id": "8a8890e1932832af019328c3ff422a94", "pin": "317009", "name": "edf", "lastName": "lee" }], }</pre>

	<pre>{ "id": "8a8890e1932832af0193288f0d531666", "pin": "5", "name": "", "lastName": "" }</pre>
Response Result Description	Data: reservation participant array list; Id: person id; Pin: person pin; Name: person name; lastName: person last name;

2.46.7 Add/Edit Space Reservation

Post Request URL	http://serverIP:serverPort/api/spaceReservation/addSpaceReservation?access_token={apitoken}
Request Mode	POST
Request content	<pre>{ "id": "", "subject": "subject name", "applyPersonEmail": "xxxx@zkteco.com", "applyPersonPin": 1234567,</pre>

	<pre> "content": "Please enter the content", "endTime": "2024-12-06 18:50:19", "participantIds": "8a8890e1932832af0193288ed42f162a,8a8890e1932832af0193288ee1251634", "remindTime": "15m", "serviceIds": "8a8890e192b76ffd0192b7718036373d,8a8890e192b76ffd0192b771803e374e", "spaceId": "8a8890e193956b62019395b01162016f", "startSignInTime": 15, "startTime": "2024-12-06 17:50:19", } </pre>
Request Parameter Description	Id: reservation id, empty when adding, not empty when editing; content: content; applyPersonPin: apply person pin(required); applyPersonEmail: apply person's email; startTime: reservation start time; endTime: reservation end time; startSignInTime: how many minutes in advance to sign in, four options: 0, 5, 15, 30 (unit: minutes); participantIds: participant ID(required), multiple IDs are separated by commas; spcaeld: reservation space id(required); serviceIds: space service ID, multiple IDs are separated by commas; 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;

Response Results	<pre>{ "code": 0, "message": "success", "data": null }</pre>
Response Result Description	Description of the generic response parameters

2.47 Parking Cost Interface

2.47.1 Parking Vehicle Billing

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

	<pre> "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>
Response Result Description	carNumber: license plate; recordId: entrance sign; carAuth: temp; finalAmount: fee amount; chargeType: tempCharge; checkInTime: admission Time; parkingTime: parking duration;

2.47.2 Parking Vehicle Payment Result Notification

Post Request URL	http://serverIP:serverPort/api/v1/parkCost/payNotify&access_token={apitoken}
Request Mode	POST
Request	{

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

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

