



MINNO Smart Locker

Internet API

V5.10

VERSION INFO

NO	VERSION	DATE	MAKE BY	Content
1	V5.05	2023-2-14	HONGWEI LIU	Add the 39th interface
2	V5.06	2023-8-1	HONGWEI LIU	Add the 40th interface and add a public interface for unlocking, powering on, and powering off (including accessories) without a screen
3	V5.07			
4	V5.08	2023-11-21	DAOWEI ZHANG	Organize documents and directories

content

Content.....	2
1. interface for pack-up and unpacking.....	4
2. status and access record query interface.....	4
3.open/close lockers,switth on/off power	6
4. get status by device number	7
5. android host uploads device abnormal information at fixed time	8
6. post after parcel deposited	8
7. post after parcel collected.....	9
8. common operation box interface	9
9. public drop off interface	10
10. public query cargo interface.....	10
11. public get/input interface	12
12. public query document quantity interface	13
13. create a new user interface	14
14. modify user informtion interface	14
15. public query device list interface.....	15
16. public query cargo type list interface	17
17. smart laundry locker order interface.....	19
18. register customer information interface	20
19. get currency list interface.....	20
20. upload product image interface	22
21. upload smart locker snapshot picture interface	23
22. get smart locker picture path interface	24
23. get push failure record interface	25
24. get the device information of all equipment in the site.....	27
25. get device total box quantity information	29
26. obtain site information acccording to site type	30
27. save user's phone number and openid interface.....	32
28. get the phone number according to the order number	33
29. interface for querying device online status	33
30. admin clear box	33
31. remote weighing.....	34

32. upload face picture	35
33. save face pictuer	36
34. samart parcel locker users'face assign	36
35. query warehousing records.....	37
36. query outbound records	38
37. verify the order	39
38. binding cargo compartments	40
39. custom pickup code delivery	41
40. open/close lockers,switch on/off power (included accessories)	42
41. public interface for obtaining rfid data from screen devices.....	43
42. public interface for obtaining electronic control board data from screen devices	44
43. public interface for obtaining weighing board data from screen devices.....	45

1. Interface for pack-up and unpacking

- 1) Request URL: http://ServerIP:8091/upus_APP/app/expressbox/hpickupbill
- 2) Request Methods: Get
- 3) Example: http://ServerIP:8091/upus_APP/app/expressbox/hpickupbill?compno=87690001&devno=JD88&billno=FCB51245121&telno=13784521465&sign=0c2fed585ca54052d72df86e2fb1673e
- 4) Parameter Description
 - a) compno: Express firm ID
 - b) devno: Device ID
 - c) billno: Tracking number
 - d) telno: Consignee's phone number
 - e) sign: Signature, Format: compno + devno + billno + telno + token, then do md5
 - f) Return result
 - g) Return mode: json
 - h) Json: Parameter after analysis
 - i. msg: Return message, If the call fails, it is not null, success is null
 - ii. data: Return operation data, if it fails, it is null, Success is not null
 - iii. success : Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result{

```
"msg": "Signature verification failed!",
"data": null,
"success": "0"
}
```

Example of returning successful result{

```
"msg": "operate successfully!",
"data": "Pick up successfully!"
"success": "1"
}
```

2. Status and access record query interface

- 1) Request URL: http://ServerIP:8091/upus_APP/app/expressbox/billlst
- 2) Request Methods: Get
- 3) Example: http://ServerIP:8091/upus_APP/app/expressbox/billlst?compno=87690001&billno=1806130001&state no=0&fixno=A01&depuser=ADMIN&depbegtime=2018-06-13 00:00:00&dependtime=2018-06-13 23:59:59&takebegtime=2018-06-13 00:00:00&takeendtime=2018-06-13 23:59:59&recphone=18745102145&pagesize=10&pageindex=1&sign=0c2fed585ca54052d
- 4) Parameter Description
 - a) compno: Express firm ID

- b) billno: Tracking number
 - c) stateno: Parcel status
 - i. 0: Stored in locker
 - ii. 1: Pick-up by consignee
 - iii. 2: Overdue and not pick-up
 - iv. 3: Overdue but pick-up
 - d) fixno: Container number
 - e) depuser: Sender number
 - f) depbegtime: Delivery parcel start time
 - g) dependtime: Deadline of deliver parcel
 - h) takebegtime: Beginning time of pick-up parcel
 - i) takeendtime: Deadline of pick-up parcel
 - j) recphone: Consignee's phone number
 - k) expno: Internal system tracking number
 - l) pagesize: (Pagination) How many records per page
 - m) pageindex: (Pagination) Page number
 - n) sign: Signature, Format: compno + billno + stateno + fixno + depuser + token, then do md5
- 5) Return result
- a) Return mode: json
 - b) Json Parsed parameter
 - i. msg: Return message, If the call fails, it is not null, success is null
 - ii. data: Return operation data, if it fails, it is null, Success is not null JSON ARRAY
 - iii. success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
  "msg": "Parameter incorrect!",
  "data": null,
  "success": "0"
}
```

Example of returning successful result {

```
"msg": "Query success ",
"data": {
  "pagecnt": "1",
  "reccnt": "1",
  "record": [
    {
      "recno": "1",
      "deptel": "13633880292",
      "fixna": "x1148",
      "imgtype": "Text message sent ",
      "takecode": "890920",
      "rowid": "1",
      "takepics": "http://www.upus.cn:8091/Userfile/Attach/1568167112194-86977.jpg",
        // Pick up the snapshot image path
      "statena": "pick up by addressee ",

```

```

        "depuser": "xll",
        "taketime": "2019-09-11 09:58:31.937",
        "deptime": "2019-09-11 09:58:11.02",
        "recphone": "13101794443",
        "expno": "67066334223928",
        "shelfna": "005048A11",
        "billno": "jgd",
        "recname": null,
        "deppics": "http://www.upus.cn:8091/Userfile/Attach/1568167092443-41258.jpg"
                // Delivery parcel Snapshot image path
        "takeman": "Name of addressee",
        "depname": " Depositor ",
    }
}
],
},
"success": "1"
}

```

3. Open/close lockers, switch on/off power

1. Request URL: http://ServerIP:8091/upus_APP/app/Lock/openLocks
2. Request Methods: Get
3. Example:

[http://ServerIP:8091/upus_APP/app/Lock/openLocks?compno=87690001&logno=ADMIN&action=0&data={\"fixno\":\"00201\", \"dtl\":{\"shelfno\":\"00201A101\", \"boardno\":\"1\", \"lockno\":\"1\"}, {\"shelfno\":\"00201A102\", \"boardno\":\"1\", \"lockno\":\"2\"}}&sign=0c2fed585ca54052d](http://ServerIP:8091/upus_APP/app/Lock/openLocks?compno=87690001&logno=ADMIN&action=0&data={\)

 - 1) Parameter Description
 - 2) compno: Express firm ID
 - 3) action: Button selection: 0 open lock 1 switch on 2switch off 3 status
 - 4) logno: User logged in
 - 5) sign: Signature, Format: compno + logno+ action+token, then do md5
 - 6) data: json data, fixno Device ID, dtl Detail data, shelfno locker id, boardno board number, lockno lock number
5. Return result
 - 1) Return mode: json
 - 2) Json Parsed parameter
 - i. msg: Returned message !
 - ii. sucess: Call success or failure flag, "1" means success, "0" means failure

```

Example of returning failed result{
  "msg": "cabinet network not reliable!",
  "success": "0"
}

```

Example of returning successful result

```
{
"success": "1",
"msg": "Locker or Locker Channel A02: OPENED"
}
```

4. Get status interface by device number

1. Request URL: http://ServerIP:8091/upus_APP/app/expressbox/allstate
2. Request Methods : POST, The request parameter is placed into the request stream
3. Example of request parameter: {"compno": "87695551", "devno": "A01", "sign": "0c2fed585ca54052d"}
4. Parameter Description
 - 1) compno: Express firm ID
 - 2) devno: Device ID
 - 3) sign: Signature, Format: compno + devno + token, then do md5
5. Return result
 - 1) Return mode: json
 - 2) Json Parsed parameter
 - a) Msg: Return message, If the call fails, it is not null, success is null !
 - b) data: Return operation data, if it fails, it is null, Success is not null. when it is successful, it is json array according to board number, success in array is status of return data, 0 means failure, 1 means success, msg is status, it is binary data listed by lock number, Electronic lock with door opening feedback , 0 means open, 1 means close, Electronic lock with closed door feedback is the opposite
 - c) success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
"msg": "Signature verification failed!",
"data": null,
"success": "0"
}
```

Example of returning successful result

```
{
"msg": "Successful operation!",
"data": [
{
"boardno": 1,
"success": "1",
"msg": "10111111111111111111111111111111"
},
{
"boardno": 2,
"success": "0",
```

```

    "msg": "query timeout"      }
  ],
  "success": "1"
}

```

5. Android host uploads device abnormal information at fixed time

1. Request URL: http://ServerIP:8091/upus_APP/app/expressbox/uploadererror
2. Request Methods: POST, The request parameter is placed into the request stream
3. Request parameter sample : {"compno": "87695551","devno": "A01","shelfno": "A01","sign": "0c2fed585ca54052d"}
4. Parameter Description
 - 1) compno: Express firm ID
 - 2) devno: Device ID
 - 3) shelfno: Locker ID
 - 4) sign: Signature, Format: compno + devno + shelfno + token, then do md5
5. Return result
 - 1) Return mode: json
 - 2) Json Parsed parameter
 - a) msg: Return message, If the call fails, it is not null, success is null
 - b) data: Return operation data, if it fails, it is null, Success is not null
 - c) success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```

{
  "msg": "Signature verification failed!",
  "data": null,
  "success": "0"
}

```

Examples of returning successful result

```

{
  "msg": "operate successfully!",
  "data": "operate successfully!",
  "success": "1"}

```

6. POST after Parcel deposited

1. POST URL: http://serverIP:serverPort/upus_APP/app/webOrder/savePartUpdateOrderState
2. Request Methods: POST, The request parameter is placed into the request stream
3. Request parameter sample: {"order Id": "original order number _receiver number", "close Time": "deposit time", "password": "verification code",

```
"location": "deposit address",
"locker": "device number",
"number": "box number"}
```

7. Post after parcel collected

- 1) Post URL: http://serverIP:serverPort/upus_APP/app/webOrder/getPartUpdateOrderStat
- 2) Request Methods: POST, The request parameter is placed into the request stream
- 3) Request parameter sample: {


```
"order Id": "original order number",
"time": "collect time",
"role": "employer",
"expno": "parcel tracking number",
"compno": company ID,
"devno": device ID}
```

8. Common operation box interface

1. Request URL: http://ServerIP:8091/upus_APP/app/cabinet/openboxes/put
2. Request method: PUT, The request parameter is placed into the request stream
3. 请求参数示例: {"compno": "87695551","devno": "A01","detail": [{"boardno": "1","lockno": "1","action": "0"},{"boardno": "2","lockno": "2","action": "0"}],"sign": "0c2fed585ca54052d"}
4. 参数说明
 - 1) compno: Express firm ID, required
 - 2) devno: Device ID, required
 - 3) boardno: Board number, required
 - 4) lockno: Lock number, required
 - 5) action: operating mode 0: open lock 1: switch on 2:switch off, required
 - 6) sign: Signature, Format: compno + devno +token, then do md5 , required
5. Return result
 - 1) Return method: json
 - 2) Json Parameter after analysis
 - a) msg Return message, If the call fails, it is not null, success is null
 - b) success: Call success or failure flag, "1" means success, "0" means failure
 - c)

Example of returning failed result

```
{
"msg": "Signature verification failed!",
"success": "0"
}
```

Example of returning successful result

```
{
```

```
"msg": " operation Successfully!",
"success": "1" }
```

9. Public drop off interface

1. Post URL: http://ServerIP:8091/upus_APP/app/cabinet/express/post
2. Post method: POST, The request parameter is placed into the request stream
3. Post parameter example: {"compno": "87695551", "devno": "A01", "logno": "ADMIN", "data": [{"shelfno": "A01", "billno": "SB18050001", "tel": "13500010001"}], {"shelfno": "A01", "billno": "SB18050001", "tel": "13500010001"}], "sign": "0c2fed585ca54052d"}
4. Parameter description
 - 1) compno: Express firm ID, required
 - 2) devno: device number, required
 - 3) logno: log in number, required
 - 4) shelfno: box number, required
 - 5) billno: original order number, required
 - 6) tel: phone number, If it has been recorded in the pre-entry, can leave it blank, otherwise it is required
 - 7) sign: Signature, format: compno + devno + logno + token, that do md5
5. Return result
 - 1) Return method: json
 - 2) Json: Parameter after analysis
 - a) msg: Return message, If the call fails, it is not null, success is null
 - b) success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
"msg": "Signature verification failed!",
"success": "0"
}
```

Example of returning successful result

```
{
"msg": " operation Successfully!",
"success": "1"
"data": [{"shelfno": "A01", "billno": "SB18050001", "tel": "13500010001", "takecode": "124568"}, {"shelfno": "A01", "billno": "SB18050001", "tel": "13500010001", "takecode": "845541"}]
}
```

10. Public query cargo interface

1. Request URL: http://ServerIP:8091/upus_APP/app/cabinet/boxes/get
2. Request method: GET, The request parameter is placed into the request stream

3. Request parameter example:
?compno=87695551&devno=A01&boxtype=1&cateno=2&sign=0c2fed585ca54052d
4. Parameter description
 - 1) compno: Express firm ID, required
 - 2) devno: device ID, required
 - 3) boxtype: box type 0 can be used 1 default all
 - 4) cateno: bill type 0 express 1 parcel 2 default no charge
 - 5) sign: Signature, format: compno + devno +token, then do md5
5. Return result
 - 1) Return method: json
 - 2) Json: parameter after analysis
 - a) msg: Return message, If the call fails, it is not null, success is null
 - b) data: kindno: box type no, kindna: box type name, price: storage price, stateno: status number, statena: status name, shelfno: box number, posno:box name, boardno board number , lockno: lock number
 - c) success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
  "msg": "Signature verification failed!",
  "success": "0"
}
```

Example of returning successful result

```
{
  "msg": "query successful",
  "data": [
    {
      "kindno": "2",
      "kindna": "big box",
      "price": "0.00",
      "stateno": "1",
      "lockno": "1",
      "posno": "A101",
      "shelfno": "00101A101",
      "boardno": "1",
      "statena": "cab be used"
    },
    {
      "kindno": "0",
      "kindna": "small box ",
      "price": "0.00",
      "stateno": "1",
      "lockno": "2",
      "posno": "A102",
      "shelfno": "00101A102",

```

```

"boardno": "1",
"statena": "can be used"
}
],
"success": "1"
}

```

11.Public get /input interface

1. Request URL: http://ServerIP:8091/upus_APP/app/cabinet/order/post
2. Request method: POST, The request parameter is placed into the request stream
3. Example : {"compno": "87690001","depotno": "006","logno": "ADMIN", "billno": "435233124456745","telno": "13789451245","sign": "0c2fed585ca54052d"}
4. Parameter specification
 - 1) compno: Express firm ID, required
 - 2) depotno: **warehouse number must be provided by us**
 - 3) logno: Log In ID , required
 - 4) billno: original bill number, required
 - 5) telno: phone number, required
 - 6) sign: Signature, format: compno + depotno + billno + telno +token, then do md5
5. Return result
 - 1) Return result: json
 - 2) Json: Parameter after analysis
 - i. msg: Return message, If the call fails, it is not null, success is null
 - ii. success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```

{
"msg": "Signature verification failed!",
"success": "0"
}

```

Example of returning successful result

```

{
"msg": " operation successfully!",
"success": "1"
}

```

12.Public query document quantity interface

1. Request URL: http://ServerIP:8091/upus_APP/app/cabinet/order/get
2. Request method: Get,the request parameter is placed into the request stream
3. Example: ?compno=87695551&depotno=A01&begdate=2018-01-01&enddate=2018-10-01&sign=0c2fed585ca54052d
4. Parameter specification
 - a) compno: Express firm ID, required
 - b) depotno: warehouse number must be provided by us
 - c) begdate: start date Format 2018-01-01 Required
 - d) enddate: end date Format 2018-01-01 Optional
 - e) sign: Signature, Format: compno + depotno +token then do md5
5. Return result
 - a) Return method: json
 - b) Json: Parameter after analysis
 - i. msg: Return message, If the call fails, it is not null, success is null !
 - ii. data: entry Order total amount get
delivery Total amount deposited
takeout Total amount collected
 - iii. success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
  "msg": "Signature verification failed!",
  "success": "0"
}
```

Example of returning successful result

```
{
  "msg": " Query successful!",
  "data": {
    "entryOrder": "0",
    "delivery": "3051",
    "takeOut": "2689"
  },
  "success": "1"
}
```

13. Create a new user interface

1.Request URL: http://ServerIP:8091/upus_APP/app/Register/expressRegisterOut

2.Request Methods: Get

3.Example: ?data={"cocode":"ZY","logno":"666","password":"1","cellphone":"15323457975","openid":"00201A102","userna":"bbb","custno":"002","depotno":"001","cardno":"123456","email":"123456@qq.com"}&sign=77a28602987c4b0a41c59faa8c19c10d

4.Parameter Description

a) sign: Signature, Format: logno+ cocode+ cellphone+token, then do md5

b) data: json DATA

i. cocode: Company Code, required

ii. logno: Registrant login number, required

iii. password: Login password, required

iv. cellphone: Registrant Phone, required

v. openid: Wechat openid, required

vi. userna: Name of Registrant

vii. custno: Customer Number

viii. depotno: **site number**

ix. cardno: Registrant card number

x. email: email address

5. Return result

a) Return mode: json

b) Json Parsed parameter

c) msg: Returned message !

d) sucess: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
  "msg": " This user is already registered !",
  "success": "0"
}
```

Example of returning successful result

```
{
  "success": "1",
  "msg": "Registration success!"
}
```

14.Modify user information interface

1.Request URL: http://ServerIP:8091/upus_APP/app/Register/expressUpdateUserOut

2. Request Methods: Get

3.Example: ?data={"password":"1","cellphone":"15323457975","openid":"00201A102","userna":"bbb","cardno":"123456","email":"123456@qq.com"}&cocode=ZY&logno=666&depotno=001&sign=77a28602987c4b0a41c59faa8c19c10d

4. Parameter Description

- a) sign: Signature, Format: logno+ cocode+ depotno+token, then do md5
- b) data: json DATA
 - i. password: User password
 - ii. cellphone: User mobile phone number
 - iii. openid: User Wechat openid
 - iv. userna: User name
 - v. cardno: User card number
 - vi. email: User email
- c) cocode: Company Code, required
- d) logno: User login number, required
- e) depotno: Site number, required

5. Return result

- a) Return mode: json
- b) Json Parsed parameter
 - i. msg: Returned message !
 - ii. success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
  "msg": " Failed to modify user information!",
  "success": "0"
}
```

Example of returning successful result

```
{
  "success": "1",
  "msg": " Successfully modify user information!"
}
```

15. PUBLIC QUERY DEVICE LIST INTERFACE

- 1) Request URL: http://ServerIP:8091/upus_APP/app/Laundry/getFixList
- 2) Request method: Get, the request parameter is placed into the request stream
- 3) Request parameter example:


```
?sign=415e25350c81cbf50348a46ec1323c00&compno=87550088&custno=0001&depotno=001
```
- 4) Parameter description
 - a) compno: company id, required
 - b) custno: Customer number, required
 - c) depotno: Site number, required
 - d) sign: Signature, format: compno + custno+token, then do md5
- 5) Return result
 - a) Return method: json
 - b) Json: Parameter after analysis

i. msg: [Return message, If the call fails, it is not null, success is null !

ii. data:

- depotna: Site name
- depotno: Site number
- libno: location number
- libna: location name
- fixno: device number
- fixna: device name

iii. success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
  "msg": "Signature verification failed!",
  "success": "0"
}
```

Example of returning successful result

```
{
  "msg": " search successful!",
  "data": [
    {
      "depotna": "Maritime University",
      17
      "depotno": "001",
      "libno": "00101",
      "fixna": " Large catering cabinet No. 1 ",
      "fixno": "HB01-01",
      "libna": " Nanyuan Building Entrance "
    },
    {
      "depotna": "Maritime University",
      "depotno": "001",
      "libno": "00104",
      "fixna": "Small catering cabinet No. 1",
      "fixno": "HB02-01",
      "libna": " Postgraduate entrance on the 25th floor"
    },
    {
      "depotna": "Maritime University",
      "depotno": "001",
      "libno": "00105",
      "fixna": " Small catering cabinet No. 2",
      "fixno": "HB02-02",
      "libna": " Men's Hostel 29th floor entrance"
    },
    {
      "depotna": "Maritime University",
      "depotno": "001",
```

```

"libno": "00106",
"fixna": " Small catering cabinet No. 3",
"fixno": "HB02-03",
"libna": " Entrance to the 52nd floor of the female dormitory"
},
{
"depotna": "Maritime University",
"depotno": "001",
"libno": "00102",
"fixna": " Large catering cabinet No. 2",
"fixno": "HB01-02",
"libna": " Male dormitory 34-35 floor middle"
},
{
"depotna": "Maritime University",
"depotno": "001",
"libno": "00107",
"fixna": " Small catering cabinet No. 4",
"fixno": "HB02-04",
"libna": " Men's Dormitory 58-59 floor entrance "
},
{
"depotna": "Maritime University",
"depotno": "001",
"libno": "00103",
"fixna": " Large catering cabinet No. 3",
"fixno": "HB01-03",
"libna": " Opposite to the newsstand "
}
},
"success": "1"
}

```

16. PUBLIC QUERY CARGO TYPE LIST INTERFACE

- 1) Request URL: http://ServerIP:8091/upus_APP/app/Laundry/getshelfType
- 2) Request method: Get, The request parameter is placed into the request stream
- 3) Request parameter example:
 ?sign=0a27ce59587abdfaba3f7dea556f910f&compno=87550088&fixno=HB01-01&depotno=001
- 4) Parameter description
 - a) compno: company id, required
 - b) fixno: device number, required

- c) depotno: Site number, required
- d) sign: Signature, format: compno + fixno+token, then do md5
- 5) Return result
 - a) Return method: json
 - b) Json: Parameter after analysis
 - i. msg: Return message, If the call fails, it is not null, success is null !
 - ii. data:
 - kindno: Cargo Type Number,
 - kindna: Cargo type name,
 - leftnum: Available cargo space,
 - allnum: Total cargo quantity,
 - price: Inventory price
 - iii. success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
  "msg": "Signature verification failed!",
  "success": "0"
}
```

Example of returning successful result

```
{
  "msg": "search successful!",
  "data": [
    {
      "leftnum": "89",
      "kindno": "0",
      "kindna": "small box",
      "price": "0.00",
      "allnum": "89"
    },
    {
      "leftnum": "55",
      "kindno": "1",
      "kindna": "medium box",
      "price": "0.00",
      "allnum": "79"
    },
    {
      "leftnum": "16",
      "kindno": "2",
      "kindna": "big box",
      "price": "0.00",
      "allnum": "18"
    }
  ]
}
```

```

"kindno": "3",
"kindna": "large box",
"price": "0.00",
"allnum": "2"
}
],
"success": "1"
}

```

17. Smart laundry locker order interface

1) Request URL: `http://ServerIP:8091/upus_APP/app/Laundry/saleToBgconv`

2) Request Methods: Get

3) Example:

```
?data={"fixno":"HB01-01","origno":"1525855","linktel":"13565658585","kindno":"1","cvtype":"0"}&
sign=77a28602987c4b0a41c59faa8c19c10d&compno=87550088&depotno=001
```

4) Parameter Description

- a) sign: Signature, Format: logno+origno+token, then do md5
- b) compno: company id, required
- c) depotno: Site number, required
- d) data: json, data
 - fixno: device number, required,
 - origno: Order number (required and unique),
 - linktel: User phone number, required,
 - kindno: Cargo type, required (0: small box, 1: medium box, 2: big box, 3: large box),
 - cvtype: order type, required(0: Save dirty clothes order, 1: Save clean clothes order),

5) Return result

- a) Return mode: json
- b) Json Parsed parameter
 - i. msg: Returned message !
 - ii. success : Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```

{"msg": " The order already exists !",
"success": "0"}

```

Example of returning successful result

```

{"success": "1",
"msg": " Order saved successfully!"}

```

18.Register customer information interface

- 1) Request URL: `http://ServerIP:8091/upus_APP/app/Register/bgcustinfoAPP`
- 2) Request Methods: `Get`
- 3) Example:


```
?sign=358a6223c2c3d71e88fe34362f2abbd3&compno=87690001&data={"custna":"dfjksajkl","spell":"aaa","compna":"dsggggg","principal":"gfg","telno":"345564566","faxno":"","email":"","http":"www.4387784378897.cn","addr":"sdfgdsggd","curno":"001","remark":"gdgsd"}
```
- 4) Parameter Description
 - a) sign: Signature, Format: logno+origno+token, then do md5 token: upus_O,difLF9406-87
 - b) compno: company id, required
 - c) data: json data,
 - i. custna: Customer abbreviation, required,
 - ii. spell: Customer old number,
 - iii. compna: Customer's full name,
 - iv. principal: Customer responsible,
 - v. telno: Customer company phone number,
 - vi. faxno: device number,
 - vii. email: email address,
 - viii. http: Homepage URL,
 - ix. addr: company address,
 - x. curno: Settlement currency number,
 - xi. remark: Remark
- 5) Return result
 - a) Return mode: `json`
 - b) Json Parsed parameter
 - i. msg: Returned message
 - ii. success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{"msg": "The mailbox name is illegal!",  
"success": "0"}
```

Example of returning successful result

```
{"success": "1",  
"msg": "Operation Success !"}
```

19.Get currency list interface

- 1) Request URL: `http://ServerIP:8091/upus_APP/app/Register/getcurno`
- 2) Request method: `Get`, The request parameter is placed into the request stream
- 3) Request parameter example: `?sign=358a6223c2c3d71e88fe34362f2abbd3&compno=87690001`
- 4) Parameter description
 - a) compno: company id, required
 - b) sign: Signature, format: compno +token then do md5 token: upus_O,difLF9406-87

5) Return result

a) Return method: json

b) Json: Parameter after analysis

i. msg: Return message, If the call fails, it is not null, success is null !

ii. data:

➤ curno: currency number,

➤ expchar: Currency code

iii. success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
  "msg": "Signature verification failed!",
  "success": "0"
}
```

Example of returning successful result

```
{
  "msg": "search successful!",
  "data": [
    {
      "expchar": "RMB",
      "curno": "001"
    },
    {
      "expchar": "USD",
      "curno": "002"
    },
    {
      "expchar": "HKD",
      "curno": "003"
    },
    {
      "expchar": "EUR",
      "curno": "004"
    },
    {
      "expchar": "NTD",
      "curno": "005"
    },
    {
      "expchar": "GRP",
      "curno": "006"
    },
    {
      "expchar": "THB",
      "curno": "007"
    }
  ]
}
```

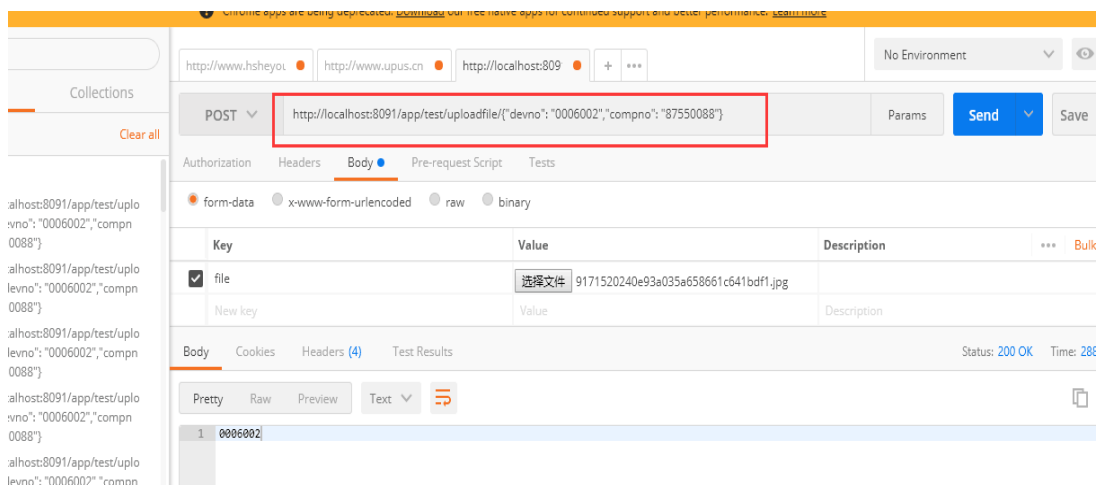
```

    {
      "expchar": "JPY",
      "curno": "008"
    },
    "success": "1"
  }

```

20.Upload product image interface

1. Request URL: `http://ServerIP:8091/upus_APP/app/BgpartManage/uploadAdvertFiles`
2. Request method: post form submission
3. Example: `http://ServerIP:8091/upus_APP/app/BgpartManage/uploadAdvertFiles/{"compno":"87690001","fixno":"JD100","pluskind":"1","boardno":"0","plusno":"0","begdate":"2018-09-06","enddate":"2018-09-30","begtime":"00:01","endtime":"23:59","stride":"1","remark":"","plusna":"","sign":"sdffdgrewre43fd"}`



4. Parameter Description

- a) compno: company id, required
- b) fixno: device number
- c) pluskind: Additional equipment or function type
 - 0: flashlight
 - 1: Cooling fan
 - 2: Half screen AD
 - 3: Full screen AD

- d) boardno: Control board number
- e) plusno: Additional equipment or function number
- f) begdate: Validity period start date
- g) enddate: Validity period end date
- h) begtime: Validity period start time
- i) endtime: Validity period end time
- j) stride: Does the end time span to the next day 0:no 1:yes
- k) remark: Remark
- l) sign: Signature, format: compno+fixno+token, then do md5

5. Return result

- 1) Return method: json
- 2) Json: Parameter after analysis
 - a) msg: Return message
 - b) success: Call success or failure flag, " 1 " means success, " 0 " means failure

Example of returning failed result

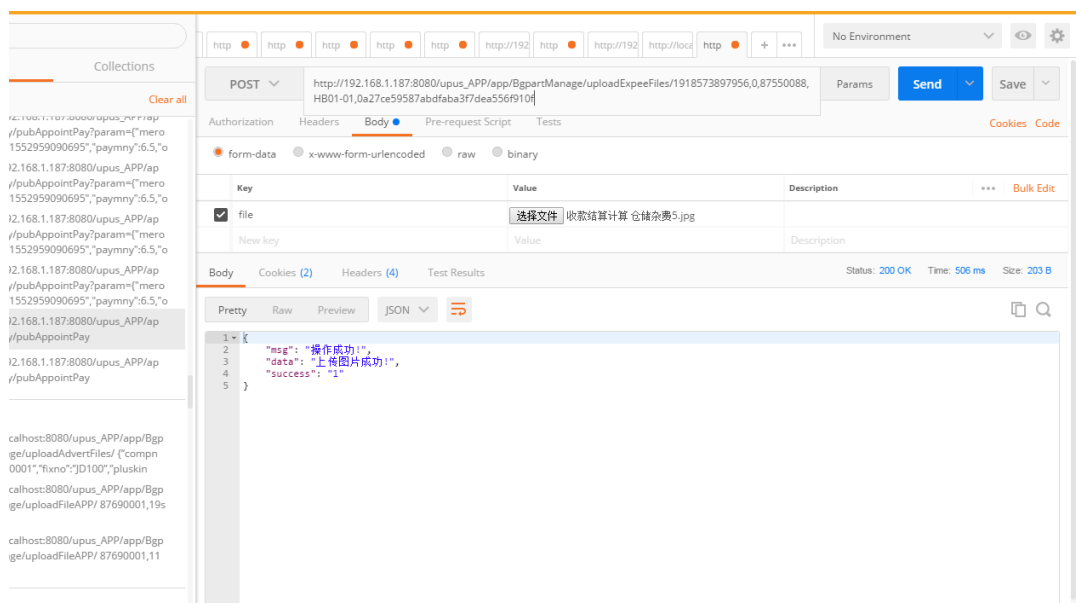
```
{ "msg": "signature verification failed!",
  "success": "0"
}
```

Example of returning successful result

```
{ "msg": "Successful operation!",
  "success": "1"
}
```

21.Upload smart locker snapshot picture interface

1. Request URL: http://ServerIP:8091/upus_APP/app/BgpartManage/uploadExpeeFiles
2. Request method: post
http://ServerIP:8091/upus_APP/app/BgpartManage/uploadExpeeFiles/1918573897956,0,87550088,H801-01,0a27ce59587abdfaba3f7dea556f910f



3. Parameter Description

Example:1918573897956,0,87550088,HB01-01,0a27ce59587abdfaba3f7dea556f910f

- ▷ 1918573897956→Original order number
- ▷ 0→drop off 1→pick up
- ▷ 87550088→company id
- ▷ HB01-01→device number
- ▷ 0a27ce59587abdfaba3f7dea556f910f →Signature, format: company id+ device number +token, then do md5

4. Return result

- a) Return method: json
- b) Json: Parameter after analysis
 - i. msg: Return message
 - ii. success: Call success or failure flag, " 1" means success, " 0" means failure

Example of returning failed result

```
{
  "msg": "signature verification failed!",
  "success": "0"
}
```

Example of returning successful result

```
{
  "msg": " Picture uploaded successfully!",
  "success": "1"
}
```

22.Get smart locker picture path interface

1) Request URL: http://ServerIP:8091/upus_APP/app/BgpartManage/getPictPath

2) Request method: get

3) Example :

http://localhost:8080/upus_APP/app/BgpartManage/getPictPath?compno=87550088&fixno=HB01-01&filename=1562140718103-17523.jpg&sign=0a27ce59587abdfaba3f7dea556f910f

4) Parameter Description

- a) compno: company id, required
- b) fixno: device number
- c) filename: Picture file name
- d) sign: Signature, format: compno + fixno+ token, then do md5

5) Return result

- a) Return method: json
- b) Json: Parameter after analysis
 - i. msg: Return message, If the call fails, it is not null, success is null !
 - ii. data: Return operation data
 - iii. success: Call success or failure flag, " 1" means success, " 0" means failure

Example of returning failed result

```
{
```

```
"msg": "signature verification failed!",
"data": null,
"success": "0"
}
```

Example of returning successful result

```
{
"success": "1",
"data": "http://ServerIP:8091/Userfile/Attach/1562140718103-17523.jpg",
"msg": "search successful!"
}
```

23. Get push failure record interface

- 1) Request URL: `http://ServerIP:8091/upus_APP/app/expressextra/pusherrors`
- 2) Request method: `get`
- 3) Example : `http://localhost:8080/upus_APP/app/expressextra/pusherrors?compno=87550088& devno=HB01-01& cateno=1& begdate=2018-08-07 00:00:00& enddate =2019-08-07 00:00:00& pagesize =10& pageindex =1&sign=0a27ce59587abdfaba3f7dea556f910f`
- 4) Parameter Description
 - a) compno: company id, required
 - b) devno: device number
 - c) cateno: Push type 0 drop off 1 pick up
 - d) begdate: Push start date
 - e) enddate: Push deadline
 - f) pagesize: The number of records displayed per page
 - g) pageindex: Page index
 - h) sign: Signature, format: compno + devno+ token, then do md5
- 5) Return result
 - a) Return method: `json`
 - b) Json: Parameter after analysis
 - i. msg: Return message, If the call fails, it is not null, success is null!
 - ii. data: Return operation data
 - iii. success: Call success or failure flag, " 1 " means success, " 0 " means failure

Example of returning failed result

```
{
  "msg": "signature verification failed!",
  "data": null,
  "success": "0"
}
```

Example of returning successful result

```
{
  "msg": "search successful!",
```

```

    "data": {
      "pagecnt": "23",
      "recnt": "46",
      "record": [
        {
          "expno": "58866892756685",
          "pushurl": "https://wxapi.itknow.cn/WechatToken/FlickNotify",
          "recno": "1",
          "pushdata": [
            {
              "orderId": "xdre_13823353929",
              "closeTime": "2019-07-16 14:35:01.247",
              "password": "792728",
              "location": "UPUS",
              "locker": "0000001",
              "number": "0000001A101",
              "expno": "58866892756685",
              "compno": "87550088",
              "devno": "0000001",
              "cvtype": ""
            }
          ],
          "response": "connect timed out",
          "fixno": "0000001",
          "createdt": "2019-07-16 14:35:11.53",
          "rowid": "1"
        },
        {
          "expno": "58866892756685",
          "pushurl": "https://wxapi.itknow.cn/WechatToken/TakenNotify",
          "recno": "1",
          "pushdata": {
            "orderId": "xdre",
            "time": "2019-07-16 14:35:24.33",
            "expno": "58866892756685",
            "compno": "87550088",
            "devno": "0000001"
          },
          "response": "connect timed out",
          "fixno": "0000001",
          "createdt": "2019-07-16 14:35:34.357",
          "rowid": "2"
        }
      ]
    },
    "success": "1"

```

```
}
```

24. Get the device information of all equipment in the site

1. Request URL: `http://ServerIP:8091/upus_APP/app/Lock/getShelfnoNum`
2. Request method: `get`
3. Example:
`http://ServerIP:8091/upus_APP/app/Lock/getShelfnoNum?compno=87550088&depotno=000&sign=b78da70c5d3f009ca1e1d9e35efa53f6`
4. Parameter Description
 - 1) `compno`: company id
 - 2) `depotno`: site number
 - 3) `sign`: Signature, format: `compno + depotno + token`, then do md5
5. Return result
 - 1) Return method: `json`
 - 2) Json: Parameter after analysis
 - a) `msg`: Return message, If the call fails, it is not null, success is null
 - b) `data`: Return operation data `depotno`: site number, `dleftnum`: Number of available site cargoes, `dallnum`: total number of site cargoes, `fleftnum`: Number of available device cargoes, `fallnum`: total number of device cargoes,
 - c) `sleftnum`: The number of specific cargo type available, `sallnum`: Total number specific cargo compartment type
 - d) `success`: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
  "msg": "signature verification failed!",
  "data": null,
  "success": "0"
}
```

Example of returning successful result{

```
"success": "1",
"data": {
  "depotno": "000",
  "dleftnum": "152",
  "dallnum": "168",
  "ddtl": [{
    "fixna": "UPUE TEST 02",
    "fixno": "0000002",
    "fdtl": [{
      "kindno": "0",
      "kindna": "small box",
      "sleftnum": "3",
      "sallnum": "9"
    }, {
      "kindno": "1",
      "kindna": "medium box",
```

```

        "sleftnum": "0",
        "sallnum": "0"
    }, {
        "kindno": "2",
        "kindna": "big box",
        "sleftnum": "0",
        "sallnum": "1"
    }, {
        "kindno": "3",
        "kindna": "large box",
        "sleftnum": "0",
        "sallnum": "0"
    }],
    "libna": "UPUS",
    "fallnum": "10",
    "fleftnum": "3"
}, {
    "fixna": "8 way 10.1inch test machine",
    "fixno": "0000009",
    "fdtl": [{
        "kindno": "0",
        "kindna": "small box ",
        "sleftnum": "8",
        "sallnum": "8"
    }, {
        "kindno": "1",
        "kindna": "medium box",
        "sleftnum": "0",
        "sallnum": "0"
    }, {
        "kindno": "2",
        "kindna": "big box",
        "sleftnum": "1",
        "sallnum": "2"
    }, {
        "kindno": "3",
        "kindna": "large box",
        "sleftnum": "0",
        "sallnum": "0"
    }],
    "libna": "Luoyang Henan",
    "fallnum": "10",
    "fleftnum": "9"
}]
},
"msg": "search successful!"

```

```
}
```

25. Get device total box quantity information

1. Request URL: `http://ServerIP:8091/upus_APP/app/Lock/getShelfnoNumToFixno`
2. Request method: `get`
3. Example :
`http://ServerIP:8091/upus_APP/app/Lock/getShelfnoNumToFixno?compno=87550088&fixno=0000001&sign=147eaeae560e692173b1b14a4344a7ed`
4. Parameter Description
 - a) `compno`: company id, required
 - b) `fixno`: device number
 - c) `sign`: Signature, format: `compno + fixno + token`, then do md5
5. Return result
 - a) Return method: `json`
 - b) Json: Parameter after analysis
 - i. `msg`: Return message, If the call fails, it is not null, success is null !
 - ii. `data`: Return operation data
 - ▷ `fixno`: device number,
 - ▷ `fixna`: device name,
 - ▷ `disablenum`: Number of disabled boxes
 - ▷ `usablenum`: Number of available boxes
 - ▷ `retainnum`: Number of reserved boxes,
 - ▷ `emplaynum`: Number of Occupied boxes,
 - ▷ `allnum`: Total number of boxes
 - iii. `success`: Call success or failure flag, " 1 " means success, " 0 " means failure

Example of returning failed result

```
{
  "msg": "signature verification failed!",
  "data": null,
  "success": "0"
}
```

Example of returning successful result

```
{
  "success": "1",
  "data": [{
    "retainnum": "0",
    "fixna": "UPUS01",
    "fixno": "0000001",
    "usablenum": "6",
    "emplaynum": "4",
    "disablenum": "0",
    "allnum": "10"
  }],
}
```

```
"msg": "search successful!"
}
```

26. Obtain site information according to site type

1) Request URL: `http://ServerIP:8091/upus_APP/app/Lock/getShelfposInfo`

2) Request method: `get`

3) Example :

```
http://ServerIP:8091/upus_APP/app/Lock/getShelfposInfo?compno=87550088&fixno=0000001&poskind=5&sign=a8241e8a644fe1b8696313c2bd353e6b
```

4) Parameter Description

a) compno: company id, required

b) fixno: device number

c) poskind: cargo type (0 small box 1 medium box 2 large box 3 large box, Pass other values to get all relevant data)

d) sign: Signature, format: compno + fixno+poskind+ token, then do md5

5) Return result

a) Return method: `json`

b) Json: Parameter after analysis

i. msg: Return message, If the call fails, it is not null, success is null!

ii. data: Return operation data

- logno: Cargo owner,
- poskind: cargo type number,
- kindna: cargo type name,
- stateno: cargo status,
- lockno: lock number,
- posno: cargo display number,
- shelfno: cargo id,
- maxqty: Maximum storage number of cargo compartment,
- boardno: control board number,
- partno: product number,
- statena: status name

iii. success: Call success or failure flag, " 1 " means success, " 0 " means failure

Example of returning failed result

```
{
  "msg": "signature verification failed!",
  "data": null,
  "success": "0"
}
```

Example of returning successful result

```
{
  "success": "1",
  "data": [{
    "logno": "a",
    "poskind": "2",
    "kindna": "large box",
```

```
"stateno": "3",
"lockno": "1",
"posno": "A101",
"shelfno": "0000001A101",
"maxqty": null,
"boardno": "1",
"partno": null,
"statena": "occupied "
}, {
"logno": "LBS01",
"poskind": "2",
"kindna": "large box",
"stateno": "3",
"lockno": "6",
"posno": "A201",
"shelfno": "0000001A201",
"maxqty": null,
"boardno": "1",
"partno": null,
"statena": "occupied "
}, {
"logno": "10010",
"poskind": "0",
"kindna": "small box",
33
"stateno": "1",
"lockno": "7",
"posno": "A202",
"shelfno": "0000001A202",
"maxqty": null,
"boardno": "1",
"partno": null,
"statena": "available"
}, {
"logno": "HB02",
"poskind": "0",
"kindna": "small box",
"stateno": "1",
"lockno": "10",
"posno": "A205",
"shelfno": "0000001A205",
"maxqty": null,
"boardno": "1",
"partno": null,
"statena": "available"
}],
```

```
"msg": "search successful!"
}
```

27. Save user phone number and openid interface

1) Request URL: http://www.upus.cn:8091/upus_APP/app/Register/saveUserInfo

2) Request Methods: Get

3) Example:

[http://www.upus.cn:8091S/upus_APP/app/Register/saveUserInfo?sign=5f76c222e17731abbc7590b4d3d60af3&data={"cellphone":"15858588888","openid":"rtftydrttytterrfgAffghh","cocode":"EMS"}](http://www.upus.cn:8091S/upus_APP/app/Register/saveUserInfo?sign=5f76c222e17731abbc7590b4d3d60af3&data={)

4) Parameter Description

a) sign: Signature, Format: openid+ cellphone+token, then do md5

b) data: json data,

- cocode: company code, required,
- cellphone: user cellphone number, required,
- openid: wechat openid, required

5) Return result

a) Return mode: json

b) Json Parsed parameter

c) msg: Returned message !

d) success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```
{
  "msg": "signature verification failed!",
  "success": "0"
}
```

Example of returning successful result

```
{
  "success": "1",
  "msg": "Operation successfully!"
}
```

28. Obtain the mobile phone number according to the tracking number

- 1) Post URL: `http://serverIP:serverPort/upus_APP/app/webOrder/getPhoneNumber`
- 2) Post Methods: POST, The request parameter is placed into the request stream
- 3) Request parameter sample:
`{"origno": "Original order number ", "logno": "Login ID ", "devno": "device number"}`
- 4) Example of returning successful result

```
{
  "success": "1",
  "msg": "Successful operation!",
  "recphone": "15696965989"
}
```

29. Interface for querying device online status

1. POST URL: `http://www.upus.cn:8091/upus_APP/app/Setting/getIsFix`
2. Post Methods: POST, The request parameter is placed into the request stream
3. Example:
`{"compno": "[company ID]", "fixno": "[device ID]", "logno": "[accession number]",
"sign": "[encryption, compno+fixno+logno+token(compno,fixno)]"}`
4. Example of returning result

```
{
  "online": "1",
  "onlinena": "on line"
}
```

30. Admin clear box

1. POST URL: `http://serverIP:serverPort/upus_APP/app/clearBox/clearLock`
2. Post Methods: POST, The request parameter is placed into the request stream
3. Example:

```
{
  "compno": "87550098", //Company ID
  "logno": "19173520241", //Login ID
  "data": {
    "fixno": "spltest", //device ID
    "dtl": //details
    [
      {
        "shelfno": "M1002" //box number
      }
    ]
  }
}
```

```

    },
    {
        "shelfno":"M1003"
    }
]
},
"sign":"8yo899"//Signature, generation method: compno + logno+token and then do md5
}

```

4. Example of returning result

```

{
    "success": "1",
    "msg": "Successful operation!"
}

```

31. remote weight

1.POST URL: http://serverIp:serverPort/upus_APP/app/checkweight/weigh

2.Request methods: POST, The request parameter is placed into the request stream

3.Example:

```

{
    "compno":"87550100-1",//company ID
    "fixno":"DV001",//device ID
    "boardno":"1",//board number
    "lockno":"2",//lock number
    "sign":"432054600320e0b33bad63939165a8b8"//Signature, generation method: compno
+fixno+ boardno + lockno +token, and then do md5.
}

```

4. Example of returning successful result

```

{
    "success":"1",
    "data":"20.05",//unit: g
    "msg":"Weighing done"
}

```

5. Example of returning failure result

```

{
    "success":"0",
    "data":null,
    "msg":"Failed to weigh"
}

```

32.Upload face image

1) Post URL: `http://192.168.1.197:8080/upus_APP/app/uploaddownloadface/uploadFilePic/{data}`

2) Post Methods: POST, The request parameter is placed into the request stream

3) Example:

{data} is:

```
{
  "compno":"87690001",//company ID
  "logno":"ADMIN",//login ID
  "sign":"4cf63fd93aa003fac2f72c0e8c42e4e5",//Signature, generation method compno +logno +token,
  do MD5"depotno":"006"//Site number, not required
}
```

File is the file parameter name, the screenshot below for details

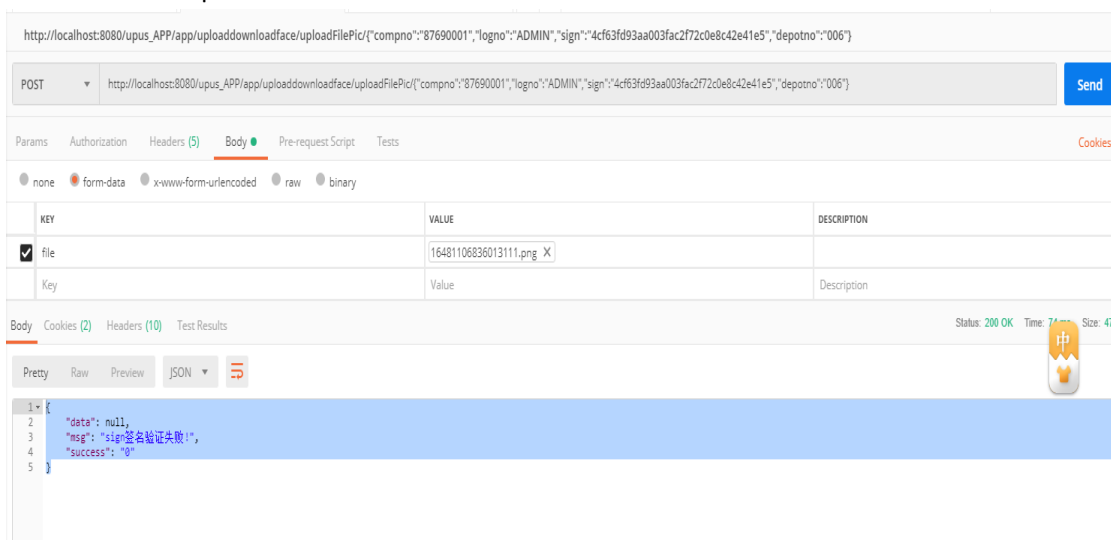
4) Example of returning successful result

```
{
  "success": "1",
  "msg": "Uploaded successfully!",
  "filename": "1649325827319146872.png"//The name of the face photo file saved by the server}
```

5) Example of returning failure result

```
{
  "data": null,
  "msg": "sign Signature verification failed!",
  "success": "0"
}
```

Screenshot example:



33. Save face image

1. POST URL: http://serverIP:serverPort/upus_APP/app/uploaddownloadface/upuserpic

2. Post Methods: POST, The request parameter is placed into the request stream

3. Example:

```
{
  "compno": "87690001", //company ID
  "logno": "ADMIN", //login ID
  "userno": "00092", //User login number for the photo
  "facepic": "1649325254913189998.png", //photo file name
  "sign": "66824a9b70740cf9174540588abc5480",
  //Signature, compno +logno+userno+facepic+token, do MD5
  "depotno": "" //Site number, not required
}
```

4. Example of returning successful result

```
{
  "data": null,
  "msg": "Successful operation!",
  "success": "1"
}
```

5. Example of returning failure result

```
{
  "data": null,
  "msg": "sign Signature verification failed!",
  "success": "0"
}
```

34. Smart parcel locker users face assign

1) Post URL: http://serverIP:serverPort/upus_APP/app/uploaddownloadface/fixreg/downdata

2) Post Methods: POST, The request parameter is placed into the request stream

3) Example:

```
{
  "compno": "87690001", //company ID
  "fixnos": [{"fixno": "JD13"}, {"fixno": "JD14"}], //device ID
  "users": [{"logno": "00092"}, {"logno": "A00002"}], //User login number
  "action": "0", //Execute action, 0 send normal data 1 send delete data
  "sign": "9f9bc2f43684cbd3aff60ce0db9ecc70", //sign,
  Generation method:
```

compno +(logno1+logno2...+lognon)+(fixno1+fixno2+...fixnon)+action+token,then do MD5

Such as : company number + login number 1+ login number 2+...+ login number n+ device number 1+

device number 2+.. device number n+ execution action+token, do MD5

"depotno":""//Site number, not required}

4) Example of returning successful result

```
{
  "data": null,
  "msg": "Successful operation!",
  "success": "1"
}
```

5) Example of returning failure result

```
{
  "data": null,
  "msg": "sign Signature verification failed!",
  "success": "0"
}
```

35.Query warehousing records

1. Post URL: http://serverIp:serverPort/upus_APP/app/bgstock/findInstk

2. Post Methods: POST, The request parameter is placed into the request stream

3. Example:

```
{
  "compno":"87550098",//company ID
  "logno":"admin",//login ID
  "sign":"9c632af6e24683e4ef28d40e934779ac",// Signature, generation method: compno + logno +
  depotno + site token, then do MD5
  "depotno":"5009",//site ID
  "fixno":"5009002",//Device ID, not required
  "pageindex":"1",//Page size
  "pagesize":"1",//paginated index
}
```

4. Example of returning successful result

```
{
  "success": "1",
  "recCnt": 13,
  "pageCnt": 13,
  "msg": "search successful!",
  "data": [
    {
      "depotna": "JIAHE",
      "fixna": "JIAHE 和 02",
      "stkdate": "2021-11-16 11:29:30.897",
      "partna": "Love medicine chest",
    }
  ]
}
```

```

    "createdt": "2021-11-16 11:29:30.897",
    "shelfno": "5009002A101",
    "posno": "A101",
    "rowid": "1",
    "depotno": "5009",
    "noteno": "F-RK2111000003",
    "fixno": "5009002",
    "creator": "5009",
    "dealer": "5009",
    "stkbat": "IB2111000021",
    "stkqty": "200",
    "partno": "500908"
  }

```

5. Example of returning failure result

```

{
  "data": null,
  "msg": "sign Signature verification failed!",
  "success": "0"
}

```

36. Query outbound records

1. POST URL: `http://serverIp:serverPort/upus_APP/app/bgstock/findOutstk`
2. Request methods: POST, The request parameter is placed into the request stream
3. Example:

```

{
  *      "compno": "87550098", //company ID
  *      "logno": "admin", //login ID
  *      "sign": "9c632af6e24683e4ef28d40e934779ac", //Signature, generation method: compno +
  *      logno + depotno + site token, then do MD5
  *      "depotno": "5009", //site ID
  *      "fixno": "5009002", //Device ID, not required
  *      "pageindex": "1", //Page size
  *      "pagesize": "1" //paginated index
  * }

```

4. Example of returning successful result

```

{
  "success": "1",
  "recCnt": 3,
  "pageCnt": 3,
  "msg": "search successful!",
  "data": [
    {
      "depotna": "JIAHE",
      "fixna": "JIAHE 和 02",

```

```

      "stkdate": "2021-10-30 10:02:36.54",
      "partna": "mineral water",
      "createdt": "2021-10-30 10:02:36.54",
      "shelfno": "5009002B106",
      "posno": "B106",
      "rowid": "3",
      "depotno": "5009",
      "noteno": "F-CK2110000671",
      "fixno": "5009002",
      "creator": "13538060903",
      "dealer": "5009",
      "stkbat": "OB2110000903",
      "stkqty": "1",
      "partno": "5009001"
    }
  ]
}

```

5. Example of returning failure result:

```

{
  "data": null,
  "msg": "sign Signature verification failed!",
  "success": "0"
}

```

37. Verify order

1. POST URL: `http://serverIp:serverPort/upus_APP/app/washerOrder/verifyOrder`
2. Request method: POST, The request parameter is placed into the request stream
3. Example

```

{
  "wordno": "0798267450361889", //order number
  "cate": "4", // Verification category
    1) already paid
    2) employee has been put back
    3) Received order
    4 )Canceled
  "user": "asdwqasd", //User login ID
  "devno": "80000005", //device ID
  "sign": "dfgd2er5td2fg1sd25" //signature
}

```

4. Example of returning successful result

```

{
  "data": {

```

```

        "shelfkind": "0", //box type
        "stateno": "5", //box status
        "shelfno": "5009001B105" //box number
    },
    "msg": "success",
    "success": "1"
}

```

5. Example of returning failed result

```

{
    "data": null,
    "msg": " Laundry order does not exist or Order status mismatch ",
    "success": "0"
}

```

38.Binding cargo compartments

1. Request URL: http://ServerIP:8091/upus_APP/app/Hyperlink/bindUser

2. Request method: Post

3. Example

```

{"compno":"87550091","logno":"ADMIN","shelfnos":"A101,A102","lognolics":"001,002","sign":"asdewas1ddas254er1xz12as"}

```

4. Parameter Description

- 1) compno: Express firm ID
- 2) logno: login ID
- 3) shelfnos: Bind box number in need, more than one box, split with “,”
- 4) lognolics: Bind the login account for users, more than one user split with “,”
- 5) sign: signature, format: MD5(compno + logno + token)

5. Return result

- 1) Return method: json
- 2) Json: Parameter after analysis
 - a) msg: Return message, If the call fails, it is not null, success is null!
 - b) data: Return operation data
 - c) success: Call success or failure flag, ” 1” means success, ” 0” means failure

Example of returning failed result

```

{
    "msg": "signature verification failed!",
    "data": null,
    "success": "0"
}

```

Example of returning successful result {

```

"msg": "operation successfully!",

```

```

"data": null,
"success": "1"
}

```

39. Custom pickup code delivery

1. Request URL: http://ServerIP:8091/upus_APP/app/newdepot/cuscode/storage
2. Request method: Post
3. Example: {"compno": "87695551", "devno": "A01", "logno": "ADMIN", "data": [{"shelfno": "A01", "billno": "SB18050001", "code": "1234"}], {"shelfno": "A01", "billno": "SB18050001", "code": "1234"}], "sign": "FGSESSSAHGSENCMJJD"}
4. Parameter Description
 - 1) compno: company ID
 - 2) logno: login ID
 - 3) devno: device number
 - 4) code: Custom code
 - 5) billno: Personal tracking number
 - 6) shelfno: box ID
 - 7) sign: 签名, 格式: MD5(compno + logno + devno + token)
5. Return result
 - a) Return mode: json
 - b) Json Parsed parameter
 - i. msg: Return message, If the call fails, success is null!
 - ii. data: Return operation data, if it fails, it is null, Success is not null
 - iii. success: Call success or failure flag, "1" means success, "0" means failure

Example of returning failed result

```

{
  "msg": "Signature verification failed",
  "data": null,
  "success": 0
}

```

Example of returning successful result

```

{
  "msg": "Operation successfully!",
  "data": null,
  "success": 1
}

```

40. Open/close lockers,switch on/off power(accessories includes)

6. Request URL: http://ServerIP:8091/upus_APP/app/hardware/shelfadd/action
7. Request Methods: Post
8. Example:{"compno":"87550010","fixno":"80000001","boardno":"1","lockno":"1","openway":"0","boardtype":"1","timeout":"","lockind":"0","action":"0","sign":"0a81dd76b0e3a2f7009f9412a9e9ac5b"}
9. Parameter Description
 - a) compno Express firm ID, Required
 - b) fixno Device ID, Required
 - c) boardno board number, Required
 - d) lockno lock number, Required
 - e) openway protocol type, Required 0-2.0 agreement 1-2.0 agree to open the motor(on) 2-2.0 agree to open the motor (off) 3-3.0 unlock agreement 4-3.0 agree to open the motor (on) 5-3.0 agree to open the motor (off)
 - f) boardtype board type, Required 0-lock board 1-5V charging board 2-DTU-WIFI 3-single- electrode control board 4-DTU-4G 5-weighing board 6-DTU-4G-V5 7-DTU-4G-V6 8-XYZ axis control board 9-motor board 10-UHF pinboard
 - g) timeout Maximum timeout time, Not required, only 3.0 agreement motor needs it
 - h) lockind feedback signal 0 open 1 close
 - i) action implementation,0 unlock 1 switch on power 2 switch off power
 - j) sign Signature, Format: MD5(compno + boardno + lockno + openway + boardtype + action + token)
10. Return result
 - a) Return mode: json
 - b) Json Parsed parameter
 - i. msg: Return message, If the call fails, success is null!
 - ii. data: Return operation data, if it fails, it is null, Success is not null
 - iii. success: Call success or failure flag,"1" means success,"0" means failure

Example of returning failed result

```
{
  "msg": "Signature verification failed",
  "data": null,
  "success": 0
}
```

Example of returning successful result

```
{
  "msg": "Operation successfully!",
  "data": null,
  "success": 1
}
```

41、Public interface for obtaining rfid data from screen devices

1. Request URL: http://ServerIP:8091/upus_APP/app/hardware/android/rfid
2. Request Methods: Post
3. Example:{"compno":"87550010","fixno":"80000001","shelfno":"1","sign":"0a81dd76b0e3a2f7009f9412a9e9ac5b"}
4. Parameter Description
 - 1) compno Express firm ID, Required
 - 2) fixno Device ID, Required
 - 3) shelfno Gird ID, Required
 - 4) sign Signature, Format: MD5(compno+shelfno+site token)
5. Return result
 - 1) Return mode: json
 - 2) Json :Parsed parameter
 - a) msg: Return message, If the call fails, success is null!
 - b) data: Return operation data, if it fails, it is null, Success is not null
 - c) success: Call success or failure flag,"1" means success,"0" means failure

Example of returning failed result

```
{
  "msg": " Signature verification failed",
  "data": null,
  "success": 0
}
```

Example of returning successful result

```
{
  "msg": "Operation successfully!",
  "data": "15184815154545,54542121212",
  "success": 1
}
```

42、Public interface for obtaining electronic control board data from screen devices

1. Request URL: http://ServerIP:8091/upus_APP/app/hardware/android/electricity
2. Request Methods: Post
3. Example:{"compno":"87550010","fixno":"80000001","shelfno":"1","sign":"0a81dd76b0e3a2f7009f9412a9e9ac5b"}
4. Parameter Description
 - 1) compno Express firm ID, Required
 - 2) fixno Device ID, Required
 - 3) shelfno Gird ID, Required
 - 4) sign Signature, Format: MD5(compno+shelfno+site token)
5. Return result
 - 1) Return mode: json
 - 2) Json :Parsed parameter
 - a) msg: Return message, If the call fails, success is null!
 - b) data: Return operation data, if it fails, it is null, Success is not null
 - c) success: Call success or failure flag,"1" means success,"0" means failure

Example of returning failed result

```
{
  "msg": "Signature verification failed",
  "data": null,
  "success": 0
}
```

Example of returning successful result

```
{
  "msg": "Operation successfully!",
  "data": {"voltage":10,"current":1,"temperature":35.8},
  "success": 1
}
```

43、Public interface for obtaining weighing board data from screen devices

1. Request URL: http://ServerIP:8091/upus_APP/app/hardware/android/weight
2. Request Methods: Post

3. Example:{"compno":"87550010","fixno":"80000001","shelfno":"1","sign":"0a81dd76b0e3a2f7009f9412a9e9ac5b"}
4. Parameter Description
 - 1) compno Express firm ID, Required
 - 2) fixno Device ID, Required
 - 3) shelfno box ID, Required
 - 4) sign Signature, Format: MD5(compno+shelfno+site token)
5. Return result
 - 1) Return mode: json
 - 2) Json :Parsed parameter
 - a) msg: Return message, If the call fails, success is null!
 - b) data: Return operation data, if it fails, it is null, Success is not null
 - c) success: Call success or failure flag,"1" means success,"0" means failure

Example of returning failed result

```
{
  "msg": " Signature verification failed",
  "data": null,
  "success": 0
}
```

Example of returning successful result

```
{
  "msg": "Operation successfully!",
  "data": "152",
  "success": 1
}
```