

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: Parcel ID
 - d) telno: Consignee's phone number
 - e) sign: Signature, Format: compno + devno + billno + telno + 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
 - 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": "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&stateno=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: Parcel ID
 - 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: Locker number
 - e) depuser: Courier ID
 - f) depbetime: Beginning time of deliver parcel
 - g) dependtime: End time of deliver parcel
 - h) takebetime: Beginning time of pick-up parcel
 - i) takeendtime: End time of pick-up parcel
 - j) recphone: Consignee's phone number
 - k) expno: Inner Record 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": " search successful!",
  "data": {
    "pagecnt": "1",
    "reccnt": "1",
    "record": [
      {
        "recno": "1",
        "deptel": "13633880292",
        "fixna": "x1148",
        "imgtype": " SMS sent",
        "takecode": "890920",
        "rowid": "1",
        "takepics": "http://www.upus.cn:8091/Userfile/Attach/1568167112194-86977.jpg",
          // Pick up the captured image path
        "statena": "Picking-up by Receiver",
        "depuser": "x11",
        "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"
        // Snapshot image path
    }
}
],
},
"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={\)
- 4) Parameter Description
 - a) compno: Express firm ID
 - b) action: Button selection:
 - i. 0 : open lock
 - ii. 1: switch on
 - iii. 2: switch off
 - iv. 3: status
 - c) logno: User logged in
 - d) sign: Signature, Format: compno + logno+ action+token, then do md5
 - e) data:
 - i. json data
 - ii. fixno: device number
 - iii. dtl: detailed data
 - iv. shelfno: locker id
 - v. boardno: board number
 - vi. lockno: lock number

5.Return result

- a) Return mode: json
- b) 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 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
 - a) compno: Express firm ID
 - b) devno: Device ID
 - c) sign: Signature, Format: compno + devno + 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. 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
 - 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": "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/ uploaderror`
- 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
 - a) compno: Express firm ID
 - b) devno: Device ID
 - c) shelfno: Locker ID
 - d) sign: Signature, Format: compno + devno + shelfno + token, then do md5
- 5) Return result
 - a) Return mode: json
 - b) JsonParsed 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
 - 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"
}
```

Examples of returning successful results

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

6. POST after Parcel deposited

- 1) Post URL: `http://` (URL provided by the customer)
- 2) Post Methods: **POST**, The request parameter is placed into the request stream
- 3) Request parameter sample: `{
 "orderId": "original order number _receiver number",
 "closeTime": "deposit time",
 "password": "verification code",
 "location": "deposit address",
 "locker": "device numbe",
 "number": "box number"
}`

7. Post after parcel collected

- 1) Post URL: `http://` (URL provided by the customer)
- 2) Post Methods: **POST**, The request parameter is placed into the request stream

- 3) Request parameter sample: {
"orderId": "original order number",
"time": "collect time"}

8. Common operation box interface

- 1) Request URL: http://ServerIP:8091/upus_APP/app/cabinet/openboxs/put
- 2) Request method: Post, The request parameter is placed into the request stream
- 3) Request parameters example: {
"compno": "87695551",
"devno": "A01",
"detail": [{"boardno": "1",
"lockno": "1",
"action": "0"},
{"boardno": "2",
"lockno": "2",
"action": "0"}],
"sign": "0c2fed585ca54052d"}
- 4) Parameter description
 - a) compno: Express firm ID, required
 - b) devno: Device number, required
 - c) boardno: Board number, required
 - d) lockno: Lock number, required
 - e) action: (required)
 - i. 0 : open lock
 - ii. 1 : switch on
 - iii. 2 : switch off
 - f) sign: Signature, Format: compno + devno + token, then do md5 , required
- 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. 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"  
}
```

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
 - a) compno: Express firm ID, required
 - b) devno: device number, required
 - c) logno: log in number, required
 - d) shelfno: box number, required
 - e) billno: original order number, required
 - f) tel: phone number, If it has been recorded in the pre-entry, can leave it blank, otherwise it is required
 - g) 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": "Successful operation!",
  "success": "1"
}
```

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
 - a) compno: Express firm ID, required
 - b) devno: device ID, required
 - c) boxtype: box type

0 :can be used,1 all, default is all

- d) cateno: bill type
 - 0 express, 1 parcel, Default no charge 2
- e) 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:
 - 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
 - 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": [
    {
      "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
 - a) compno: Express firm ID, required
 - b) depotno: Site number, required
 - c) logno: Log In ID , required
 - d) billno: original bill number, required
 - e) telno: phone number, required
 - f) sign: Signature, format: compno + depotno + billno + telno +token, then do md5
- 5) Return result
 - a) Return result: json
 - b) 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": "Successful operation!",
    "success": "1"
}

```

12. Public query receipt data

- 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 deception
 - a) compno: Express firm ID, required
 - b) depotno: depotno: Site number, required
 - 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: entryOrder 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"}&sign=77a28602987c4b0a41c59faa8c19c10d
- 4) Parameter Description
 - a) sign: Signature, Format: logno+ cocode+ cellphone+token, then do md5
 - b) data: json DATA
 - i. cocode: Company Code
 - 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
- 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": " 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"}cocode=ZY&logno=666depotno=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
 - 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 search lockers

- 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",
      "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 search lockers type

- 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"
    },
    {
        "leftnum": "1",
        "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. sucess: 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":"sdfgdsaggd","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

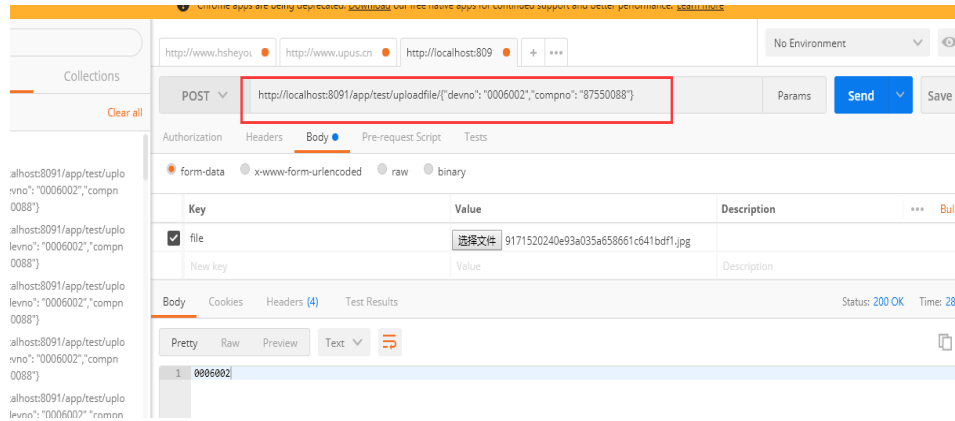
```
{
  "msg": "Signature verification failed!",
  "success": "0"
}
```

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"
}
```

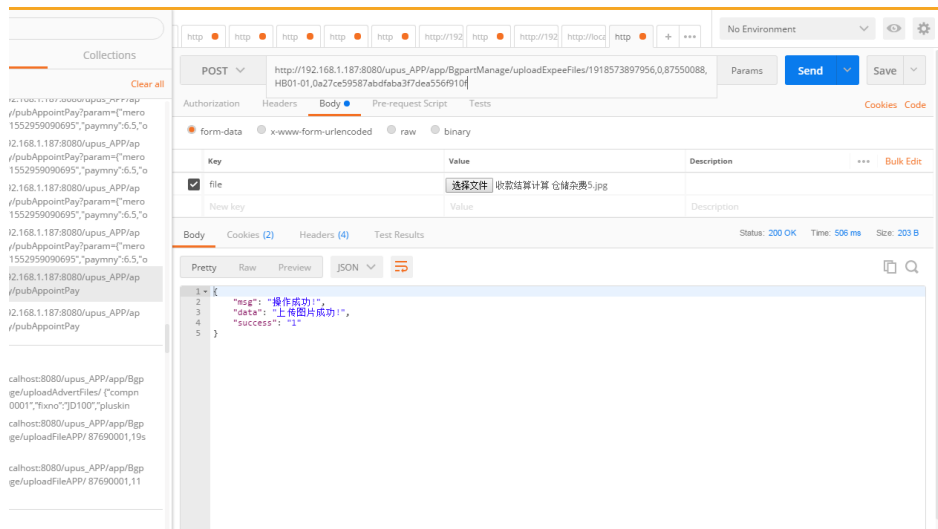
Example of returning successful result

```
{"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,HB01-01,0a27ce59587abdfaba3f7dea556f910f`



- 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",
    "reccnt": "46",
    "record": [
      {
        "expno": "58866892756685",
        "pushurl": "https://xwxapi.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://xwxapi.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 cargo 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
 - a) `compno`: company id, required
 - b) `depotno`: Site number
 - c) `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`: Return operation data
 - `depotno`: Site numbe,
 - `dleftnum`: Number of available site cargoes,
 - `dallnum`: total number of site cargoes,
 - `fleftnum`: Number of available device cargoes,
 - `fallnum`: total number of device cargoes,
 - `sleftnum`: The number of specific cargo type available,
 - `sallnum`: Specific cargo compartment type
 - iii. `sucess`: 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": "UPUS 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": " 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": "large box",
```

```

        "sleftnum": "1",
        "sallnum": "2"
    }, {
        "kindno": "3",
        "kindna": "large box",
        "sleftnum": "0",
        "sallnum": "0"
    }],
    "libna": "shenzhen",
    "fallnum": "10",
    "fleftnum": "9"
}]]
},
"msg": "search successful!"
}

```

25. Get device total cargo 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",
    "employnum": "4",
    "disablenum": "0",
    "allnum": "10"
  }],
  "msg": "search successful!"
}

```

26. Obtain cargo information according to cargo 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",  
    "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":"rtftydrtytterrfgAffghh","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": "Successful operation!"
}

```

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

- 1) Post URL: <http://>
- 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"  
}
```