alldebit

Merchant Integration Manual

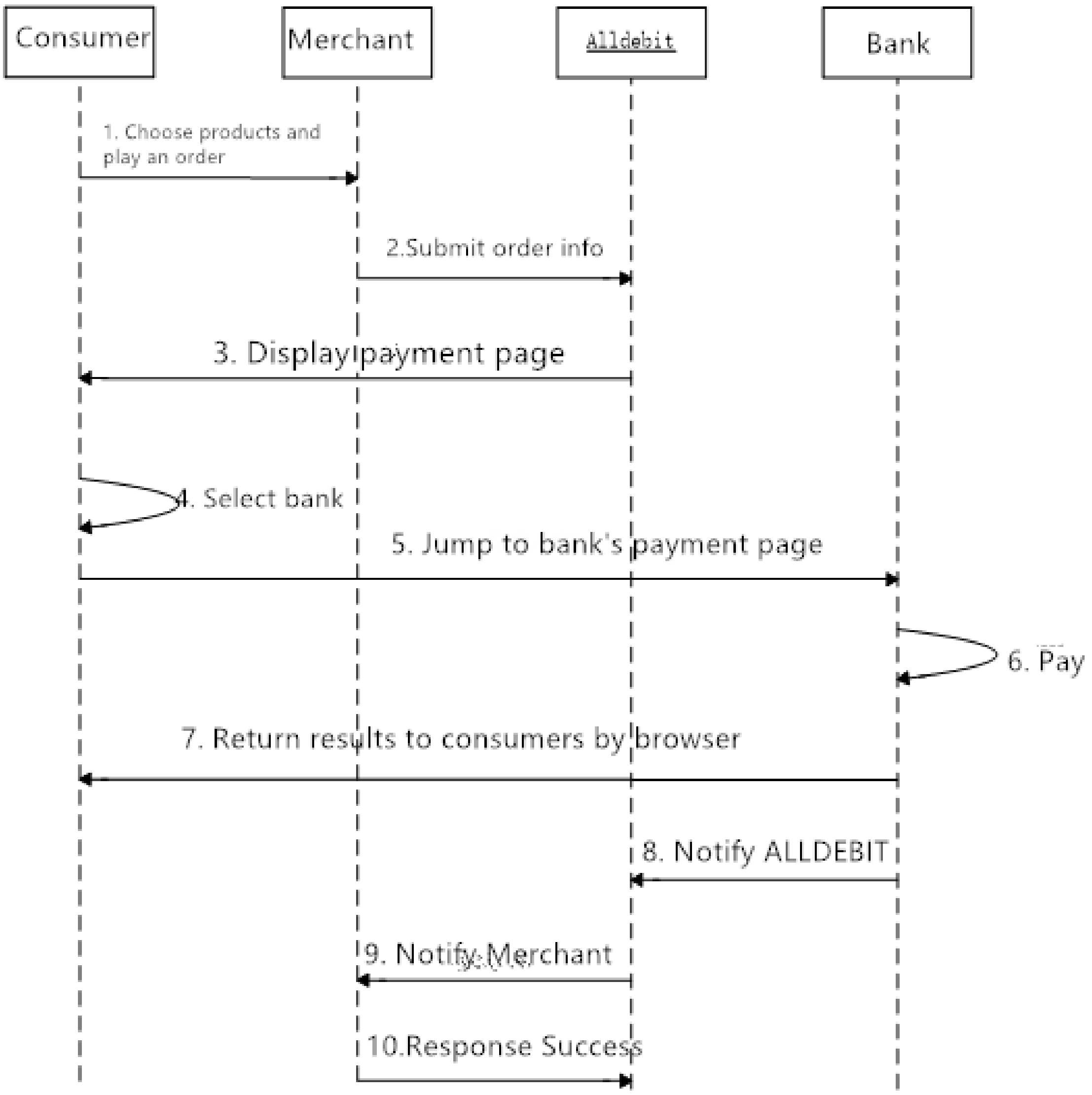
V1.2.6

1. 概述

如果是第一次集成这这个接口，建议从第一章开始阅读。如果已有类似接口的集成经验，可以跳过第一章从第二章开始阅读。

The main purpose of the Merchant Integration Manual is to provide merchant developers and system integrators with the necessary technical information for designing or programming the necessary online applications with ALLDEBIT.
The target audiences of this manual are expected to have working knowledge of the Web/Application server technology.

1.1 Transaction Process



1. Consumers choose goods and play orders on Merchants' website;客户在商户的商城上选择商品并下单。

2. Merchants submit transaction information to the gateway of ALLDEBIT; 商户将交易信息提交到 ALLDEBIT 的网关。
3. ALLDEBIT displays payment page to consumers; ALLDEBIT 向用户展示收银台支付页面。
4. Consumers choose banks on payment page; 客户在收银台支付页面选择银行。
5. Jump to Online Banking payment page; 跳转至网银支付页面。
6. Consumers use their bank account and password and U key to make payment; 使用自己的卡密和 u 盾进行付款。
7. After payment successful, the browser jumps to Merchants' website; 付款成功后，浏览器跳转至商城网站。
8. Consumers' banks notify ALLDEBIT by server; 银行通过服务器通知 ALLDEBIT。
9. ALLDEBIT notify Merchant by server; ALLDEBIT 通过服务器通知商户。
10. Merchants' website receive the server's response of success. Receiving this response means notify successfully, otherwise it will send the inquiry repeatedly, maximum in 7 times;在收到服务器返回后，在页面上输出 success。服务器收到这个响应后，认为返回成功，否则将会按照一定的时间间隔重复发送。最高发送次数为 7 次。

1.2 了解交易签名 About transaction signature

The transaction signature is a necessary measure to prevent the transaction information from being tampered during transmission. Normally use RSA in internet, both parties agree upon key strings, and prevent transaction information to be tampered through abstracting the key strings. Both parties in transaction should agree upon the key strings, and one to make abstract, the other to verify the abstract. When verify successfully, it is confirmed that the transaction information has not ever been tampered. Generally this key strings is generated by payment company, Merchant can download or update from Merchant Service backend.

Attention: Key strings can ONLY be downloaded from Merchant Service backend after merchant login, other ways of transmission are not supported.

交易签名是为了防止交易信息在传输过程中被篡改而进行的必要的措施。一般在互联网中使用 RSA，通过双方约定密钥串，并通过密钥串对交易数据进行摘要的方式来预防信息被篡改。交易双方会约定密钥串，由一方对信息做摘要，另一方对摘要进行验证，当验证匹配时，确认信息没有被篡改。一般密钥由支付公司生成，商户可以在商服后台进行下载或者更新。

注意：密钥串只能由商户登陆商户后台进行下载，不以其他任何方式传输。

2. 接口集成 Interface Integration

2.1 集成前的准备 Before Integration

a) Before developing, please prepare a domain of public net and an IP address of public net; 在开发之前请准备好公网的域名或者一个公网的 IP 地址。

b) Our technical support will send the test information including transaction IP, backend address , MID and other information to you, and help you complete the test;我们的技术支持将会将测试使用交易地址，后台地址，商户号以及其他信息发送给您，帮助您完成测试。

2.2 开发 Developing

Interface uses WebApi in restful; 接口均采用 restful 方式的 WebApi。

2.2.1 商户注册接口 Merchant Registration Interface

Pay attention to uppercase and lowercase letters of all the fields in interface;**接口中的所有字段，大小写敏感**

Address:
Action: POST
Merchant Registration Interface;商户注册接口
Live 正式地址：https://pgw.alldebitpay.com/v1/merchant/regist.json（request address of regist json; 商户注册 json 请求地址）
Test 测试地址：http://test-pgw.alldebitpay.com/merchant/regist.json（request address of regist json; 商户注册 json 请求地址）

Request Format 请求格式
application/json,text/json

Sample:

```
{
        "version":"v1.0",
        "inputCharset":"1",
        "language":"1",
        "belongOrganize":"OI20190304100054",
        "merchantNameCN":"张三测试商户",
        "merchantNameEN":"test merchant",
        "merchantIndustry":"Service trade",
        "merchantType":"1",
        "countryType":"1",
        “companyWeb”:”www.alldebit.com”,
        “companyAddress”:”xinjiapo”,
        “businessScope”:”Southeast Asia”,
        “aliasName”:”test mid”,
        “contacts”:”jack”,
        “email”:”jackman@163.com”,
        “telephone”:”13655552222”,
        “zipcode”:”518222”,
        “faxNumber”:”1234124”,
        “corporateName”:”jack”,
        “corporateIdNumber”:”456622331145226”,
        “idType”:”A”,
        “bankCode”:”ACB”,
        “branchCode”:”000”,
        “accountNo”:”000”,
        “acName”:”jack”,
        “transactionCode”:”111”,
        “beneficiaryID”:”111”,
        “beneficiaryName1”:”111”,
        “beneficiaryName2”:”222”,
        “beneficiaryName3”:”333”,
        “beneficiaryName4”:”444”,
        “beneficiaryAddress1”:”111”,
        “beneficiaryAddress2”:”222”,
        “beneficiaryAddress3”:”333”,
        “beneficiaryAddress4”:”444”,
        “beneficiaryCity”:”aaa”,
        “beneficiaryPostalCode”:”00000”,
        “beneficiaryEmail”:”123@gmail.com”,
        “countryCode”:”CHN”,
        “userName”:”lili”,
        “userEmail”:”lili@163.com”,
        “phoneNum”:”13666665555”,
        “MD5Key”:”FDF34fdsfsafgFDDSFFGS34sdf”,
        "signMsg":"ede18ee2fe0171dd0d45be9cc257a54f"
   }
```

Return Format 返回格式

application/json,text/json

```
merchantRegist：{
"version":"v1.0",
    "inputCharset":"1",
    "language":"1",
    "belongOrganize":"OI20190304100054",
    "merchantType":"1",
    "merchantId":"20190304100056",
    "loginName":"jackman",
    "loginPassword":"a123456",
    "tradePassword":"a123456",
    "respCode","T000",
    "respMsg","success",
    "sign":"ede18ee2fe0171dd0d45be9cc257a54f"
}
```

2.2.2 网银收单接口 Online Banking Acquiring Interface

Pay attention to uppercase and lowercase letters of all the fields in interface;接口中的所有字段，大小写敏感

Address:

Action: POST

Online banking acquiring interface;网银收单接口
Live 正式地址：https://pgw.alldebitpay.com/v1/beforSendToItsPay（form 表单提交）
Test 测试地址：http://test-pgw.alldebitpay.com/v1/beforSendToItsPay（form 表单提交）

Request Format 请求格式

Text/html

Sample:

```
<form action="gatewayurl" method="post">
    <input type="text" name="version" value="v1.0" />
    <input type="text" name="inputCharset" value="1" />
    <input type="text" name="language" value="1" />
    <input type="text" name="merchantId" value="100020160405003" />
    <input type="text" name="merOrderNo" value="ORD000001" />
    <input type="text" name="merorderDatetime" value="20150203101123" />
    <input type="text" name="merorderCurrency" value="CNY" />
    <input type="text" name="merorderAmount" value="100.00" />
    <input type="text" name="pickupUrl" value="http://browserrl.abcd.com" />
    <input type="text" name="receiveUrl" value="http://serverurl.abcd.com" />
    <input type="text" name="payType" value="0" />
    <input type="text" name="issuerId" value="" />
    <input type="text" name="ext1" value="系统订单号" />
    <input type="text" name="ext2" value="ext2" />
    <input type="text" name="ext3" value="ext3" />
    <input type="text" name="businessType" value="1" />
    <input type="text" name="payerName" value="张三" />
    <input type="text" name="payerEmail" value="eason@abcd.com" />
    <input type="text" name="payerTelephone" value="136112341234" />
    <input type="text" name="productId" value="s001" />
    <input type="text" name="productName" value="杭州一日行" />
    <input type="text" name="productNum" value="1" />
    <input type="text" name="productPrice" value="10000" />
    <input type="text" name="productCurrency" value="CNY" />
    <input type="text" name="merchantSignType" value="1" />
    <input type="text" name="signMsg" value="ede18ee2fe0171dd0d45be9cc257a54f" />
    <input type="submit" value="submit" />
</form>
```

Return Format 返回格式

Text/html

Sample:

```
<form action="receiveUrl" method="post">
<input type="text" name="version" value="v1.0" />
<input type="text" name="inputCharset" value="1" />
<input type="text" name="language" value="1" />
<input type="text" name="merchantId" value="100020160405003" />
<input type="text" name="merOrderNo" value="ORD000001" />
<input type="text" name="merorderDatetime" value="20150203101123" />
<input type="text" name="merorderCurrency" value="CNY" />
<input type="text" name="merorderAmount" value="100.00" />
```

```
<input type="text" name="payAmount" value="24.16" />
<input type="text" name="payCurrency" value="SGD" />
<input type="text" name="payExchangeRate" value="4.25655" />
<input type="text" name="merSTOrderAmount" value="24.16" />
<input type="text" name="merSTCurrency" value="SGD" />
<input type="text" name="exchangeRate" value="4.25655" />
<input type="text" name="ext1" value="ext1" />
<input type="text" name="ext2" value="ext2" />
<input type="text" name="ext3" value="ext3" />
<input type="text" name="tradeFee" value="5.00" />
<input type="text" name="tradeFeeCurrency" value="SGD" />
<input type="text" name="txnid" value="111502030000000008834" />
<input type="text" name="txndate" value="201502031019002" />
<input type="text" name="status" value="1" />
<input type="text" name="respcode" value="T000" />
<input type="text" name="respmsg" value="success" />
<input type="text" name="signMsg" value="d30fb34c44531874515c47ea65184a28" />
</form>
```

2.2.3 退款接口 Refund Interface

Pay attention to uppercase and lowercase letters of all the fields in interface;**接口中的所有字段，大小写敏感**

Address:
Action: POST
Refund interface of online banking 网银退款接口：
Live 正式地址：https://pgw.alldebitpay.com/v1/beforSendToItsRefund.json（request address of refund json; 退款 json 请求地址）
Live 正式地址： https://pgw.alldebitpay.com/v1/beforSendToItsRefund.xml（request address of refund xml;退款 xml 请求地址）
Test 测试地址：http://test-pgw.alldebitpay.com/v1/beforSendToItsRefund.json（request address of refund jso;退款 json 请求地址）
Test 测试地址：http://test-pgw.alldebitpay.com/v1/beforSendToItsRefund.xml （request address of refund xml; 退款 xml 请求地址）

Request Format 请求格式
application/json,text/json

Sample:

```
{
	"version":"v1.0",
	"inputCharset":"1",
	"language":"1",
	"merchantId":"100020160405003",
	"merOrderNo":"ORD000001",
	"sysOrderNo":"NT20150203101123",
	"refundAmount":"100.00",
	"refundOrdertime":"20160913121211",
	"ext1":"ext1",
	“merchantSignType”:”1”,
        "signMsg":"ede18ee2fe0171dd0d45be9cc257a54f"
 }
```

application/xml,text/xml

Sample:

```
<?xml version="v1.0" encoding="UTF-8" ?>
<requestRefund>
        <version>v1.0</version>
        <inputCharset>v1.0</inputCharset>
        <language>v1.0</language>
        <merchantId>100020160405003</merchantId>
        <merOrderNo>ORD000001</merOrderNo>
        <sysOrderNo>NT20150203101123</sysOrderNo>
        <refundAmount>100.00</refundAmount>
        <refundOrdertime>20160913121211</refundOrdertime>
        <ext1>ext1</ext1>
        <merchantSignType>1</merchantSignType>
        <signMsg>ede18ee2fe0171dd0d45be9cc257a54f</signMsg>
</ requestRefund>
```

Return Format 返回格式

application/json,text/json

```
requestRefund：{
"version":"v1.0",
      "inputCharset":"1",
      "language":"1",
      "merchantId":"100020160405003",
      "merOrderNo":"ORD000001",
      "refundAmount":"100.00",
      "refundOrdertime":"20160913121211",
      "ext1":"ext1",
      "status":"1",
      "txnId","RF20160913123212123",
      “txnDate”,”20160913123211”,
      "respCode","T000",
      "respMsg","success",
      "signMsg":"ede18ee2fe0171dd0d45be9cc257a54f"
}
```

application/xml, text/xml

```
<requestRefund>
      <version>v1.0</version>
      <inputCharset>1</inputCharset>
      <language>1</language>
      <merchantId>100020160405003</merchantId>
      <merOrderNo>ORD000001</merOrderNo>
      <refundAmount>100.00</refundAmount>
      <refundOrdertime>20160913121211</refundOrdertime>
      <ext1>ext1</ext1>
      <status>1</status>
```

```
    <txnId>RF20160913123212123</txnId>
    <txnDate>20160913123211</txnDate>
    <respCode>T000</respCode>
    <respMsg>success</respMsg>
    <signMsg>ede18ee2fe0171dd0d45be9cc257a54f</signMsg>
</ requestRefund>
```

2.2.4 订单查询接口 Order querying interface

Address:
Action: POST
Order querying interface 订单查询接口：
Live 正式环境：https://pgw.alldebitpay.com/v1/merQueryOneOrder.json（query address of json request; 查询 json 请求地址）
Live 正式环境：https://pgw.alldebitpay.com/v1/merQueryOneOrder.xml（ query address of xml request;查询 xml 请求地址）
Test 测试环境：http://test-pgw.alldebitpay.com/v1/merQueryOneOrder.json（ query address of json request;查询 json 请求地址）
Test 测试环境：http://test-pgw.alldebitpay.com/v1/merQueryOneOrder.xml（ query address of xml request;查询 xml 请求地址）

Request Format 请求格式

application/json,text/json

Sample:

```
{
    "version":"v1.0",
    "merchantId":"100020160405003",
    "merOrderNo":"ORD000001",
    "merorderDatetime":"20160913121211",
    "merchantSignType":"1",
    "signMsg":"ede18ee2fe0171dd0d45be9cc257a54f"
}
```

application/xml,text/xml

Sample:

```
<?xml version="v1.0" encoding="UTF-8" ?>
<queryOrder>
    <version>v1.0</version>
    <merchantId>100020160405003</merchantId>
    <merOrderNo>ORD000001</merOrderNo>
    <merorderDatetime>20150203101123</merorderDatetime>
    <merchantSignType>1</merchantSignType>
    <signMsg>ede18ee2fe0171dd0d45be9cc257a54f</signMsg>
</queryOrder>
```

Return Format 返回格式

application/json,text/json

```
queryOrder:{
"version": "v1.0",
"language":"1",
"inputCharset":"1",
"merchantId": "100020160405003",
"merOrderNo": " ORD000001",
"merorderDatetime": "20150203101123",
"merorderCurrency": "CNY",
"merorderAmount": "100.00",
"ext1": " remark1",
"ext2": " remark2",
"ext3": " remark3",
"txnId": "NT11150203000000008834",
"txnDate": " 201502031019002",
    “state”:”1”,
"respCode": "T000",
"respMsg": "success",
"signMsg": " d30fb34c44531874515c47ea65184a28"
}
```

application/xml, text/xml

```
<queryOrder>
<version>v1.0</version>
<language>1</language>
<inputCharset>1</inputCharset>
<merchantId>100020160405003</merchantId>
<merOrderNo>ORD000001</merOrderNo>
<merorderDatetime>20150203101123</merorderDatetime>
<merorderCurrency>CNY</merorderCurrency>
<merorderAmount>100.00</merorderAmount>
<ext1>remark1</ext1>
<ext2>remark2</ext2>
<ext3>remark3</ext3>
<txnId> NT11150203000000008834</txnId>
<txnDate>201502031019002</txnDate>
<state>1</state>
<respCode>T000</respCode>
<respMsg>success</respMsg>
<signMsg>d30fb34c44531874515c47ea65184a28</signMsg>
</ queryOrder>
```

2.2.5 指定结算接口 Appointed Settlement Interface

Address:
Action: POST
Appointed Settlement Interface 指定结算接口：
Live 正式环境：https://pgw.alldebitpay.com/OrderSettlementGW/v1/orderSettlementGW.json（json request address; json 请求地址）
Live 正式环境：https://pgw.alldebitpay.com/OrderSettlementGW/v1/orderSettlementGW.xml（xml request address;xml 请求地址）

Test 测试环境：http://test-pgw.alldebitpay.com/OrderSettlementGW/v1/orderSettlementGW.json（query address of json request;查询 json 请求地址）
Test 测试环境：http://test-pgw.alldebitpay.com/OrderSettlementGW/v1/orderSettlementGW.xml（ query ddress of xml request;查询 xml 请求地址）

Request Format 请求格式

application/json,text/json

Sample:

```
{
	"version":"v1.0",
	"inputCharset":"1",
	"language":"1",
	"bussinesType":"1",
	"refcnceFlow":"PGW000001",
	"merchantId":"201620160405003",
	"merOrderNo":"ORD000001",
	"shouldDealtime":"20170410",
	"state":"1",
	"merchantSignType":"1",
	"signMsg":"ede18ee2fe0171dd0d45be9cc257a54f"
}
```

application/xml,text/xml

Sample:

```
<?xml version="v1.0" encoding="UTF-8" ?>
<GWOrderSettlementRequest>
<version>v1.0</version>
<inputCharset>1</inputCharset>
<language>1</language>
<bussinesType>1</bussinesType>
<refcnceFlow>PGW000001</refcnceFlow>
<merchantId>100020160405003</merchantId>
<merOrderNo>100020160405003</merOrderNo>
<shouldDealtime>20170410</shouldDealtime>
<state>1</state>
<merchantSignType>1</merchantSignType>
<signMsg>ede18ee2fe0171dd0d45be9cc257a54f</signMsg>
</GWOrderSettlementRequest>
```

Return Format 返回格式

application/json,text/json

```
GWOrderSettlementRequest:{
"version": "v1.0",
“inputCharset”:”1”,
“language”:”1”,
"merchantId": "100020160405003",
"bussinesType": "1",
"refcnceFlow": "201702152028",
"merOrderNo": "100020160405003",
"shouldDealtime": "20170215",
"actualCTtime": "20170215",
"merchantSignType": "1",
        “state”:”1”,
"respCode": "T000",
"respMsg": "success",
"signMsg": " d30fb34c44531874515c47ea65184a28"
}
```

application/xml, text/xml

```
< GWOrderSettlementRequest>
<version>v1.0</version>
<inputCharset>1</inputCharset>
<language>1</language>
<merchantId>100020160405003</merchantId>
<bussinesType>1</bussinesType>
<refcnceFlow>201702152028</refcnceFlow>
<merOrderNo>100020160405003</merOrderNo>
<shouldDealtime>20170215</shouldDealtime>
<actualCTtime>20170215</actualCTtime>
<merchantSignType>1</merchantSignType>
<state>1</state>
<respCode>T000</respCode>
<respMsg>success</respMsg>
<signMsg>d30fb34c44531874515c47ea65184a28</signMsg>
</ GWOrderSettlementRequest>
```

2.2.6 移动支付接口 Mobile Payment Interface

注意：目前移动网关和 online 网关已经合并，此网关废弃，后面不在接入商户

Address:
Action: POST
Mobile Payment Interface 移动支付接口：
Live 正式环境： https://pgw.alldebitpay.com/v1/mobilePayIts（form 表单提交）
Test 测试环境： http://test-pgw.alldebitpay.com/v1/mobilePayIts（form 表单提交）

Request Format 请求格式

Text/html

Sample:

```
<form action="xxxxxUrl" method="post">
<input type="text" name="version" value="v1.0" />
<input type="text" name="inputCharset" value="1" />
<input type="text" name="language" value="1" />
<input type="text" name="merchantId" value="100020160405003" />
<input type="text" name="merOrderNo" value="ORD000001" />
<input type="text" name="merorderDatetime" value="20150203101123" />
<input type="text" name="merorderCurrency" value="CNY" />
<input type="text" name="merorderAmount" value="100.00" />
<input type="text" name="pickupUrl" value="pickupUrl" />
```

```
<input type="text" name="receiveUrl" value="receiveUrl" />
<input type="text" name="ext1" value="ext1" />
<input type="text" name="ext2" value="ext2" />
<input type="text" name="ext3" value="ext3" />
<input type="text" name="businessType" value="1" />
<input type="text" name="payType" value="1" />
<input type="text" name="issuerId" value="0" />
<input type="text" name="payerAcctNo" value="11111" />
<input type="text" name="termId" value="11111111" />
<input type="text" name="productName" value="杭州一日行" />
<input type="text" name="merchantSignType" value="1" />
<input type="text" name="signMsg" value="d30fb34c44531874515c47ea65184a28" />
</form>
```

Return Format 返回格式

Text/html

Sample:

```
<form action="xxxxxUrl" method="post">
<input type="text" name="version" value="v1.0" />
<input type="text" name="inputCharset" value="1" />
<input type="text" name="language" value="1" />
<input type="text" name="merchantId" value="100020160405003" />
<input type="text" name="merOrderNo" value="ORD000001" />
<input type="text" name="merorderDatetime" value="20150203101123" />
<input type="text" name="merorderCurrency" value="CNY" />
<input type="text" name="merorderAmount" value="100.00" />
<input type="text" name="payAmount" value="24.16" />
<input type="text" name="payCurrency" value="SGD" />
<input type="text" name="payExchangeRate" value="4.25655" />
<input type="text" name="merSTOrderAmount" value="24.16" />
<input type="text" name="merSTCurrency" value="SGD" />
<input type="text" name="exchangeRate" value="4.25655" />
<input type="text" name="ext1" value="ext1" />
<input type="text" name="ext2" value="ext2" />
<input type="text" name="ext3" value="ext3" />
<input type="text" name="tradeFee" value="5.00" />
<input type="text" name="tradeFeeCurrency" value="SGD" />
<input type="text" name="txnid" value="11150203000000008834" />
<input type="text" name="txndate" value="201502031019002" />
<input type="text" name="status" value="1" />
<input type="text" name="respcode" value="T000" />
<input type="text" name="respmsg" value="success" />
<input type="text" name="signMsg" value="d30fb34c44531874515c47ea65184a28" />
</form>
```

2.2.7 下发接口 Agent Interface

Pay attention to uppercase and lowercase letters of all the fields in interface;**接口中的所有字段，大小写敏感**

Address:
Action: POST
Refund interface of online banking 网银退款接口：
Live 正式地址：https://pgw.alldebitpay.com/agentPay.json（request address of agent json; 下发 json 请求地址）
Live 正式地址： https://pgw.alldebitpay.com/agentPay.xml（request address of agent xml; 下发 xml 请求地址）
Test 测试地址：http://test-pgw.alldebitpay.com/agent.json（request address of agent json;下发 json 请求地址）
Test 测试地址：http://test-pgw.alldebitpay.com/agent.xml （request address of agent xml; 下发 xml 请求地址）

Request Format 请求格式

application/json,text/json

Sample:

```
{
        "version":"v1.0",
        "merchantId":"100020160405003",
        "agentAmt":"100.23",
        "currencyCode":"CNY",
        "agentDatetime":"20170809152300",
        “accountNo”:”6225887836361212”,
        “accountName”:”张三”,
        “accountType”:”2”,
        “bankName”:”招商银行”,
        “bankCode”:”cmb”,
        “outAgentNo”:”OR20170505112211”,
        “bankNo”:”111111111111”,
        “ext1”:”111”,
        “ext2”:”222”,
        “agentNotifyUrl”:”http://alldebit.test.com/notify.do”,
        “merchantSignType”:”1”,
        "signMsg":"ede18ee2fe0171dd0d45be9cc257a54f"
 }
```

application/xml,text/xml

Sample:

```
<?xml version="v1.0" encoding="UTF-8" ?>
<requestRefund>
        <version>v1.0</version>
        <merchantId>100020160405003</merchantId>
        <agentAmt>100.23</agentAmt>
        <currencyCode>CNY</currencyCode>
        <agentDatetime>20170505121211</agentDatetime>
        <accountNo>6225887836361212</accountNo>
        <accountName>张三</accountName>
        <accountType>2</accountType>
        <bankName>招商银行</bankName>
        <bankCode>cmb</bankCode>
        <outAgentNo>OR20170505112211</outAgentNo>
        <bankNo>11111111111</bankNo>
        <ext1>111</ext1>
        <ext2>222</ext2>
        <agentNotifyUrl>http://alldebit.test.com/notify.do</agentNotifyUrl>
        <merchantSignType>1</merchantSignType>
        <signMsg>ede18ee2fe0171dd0d45be9cc257a54f</signMsg>
</ requestRefund>
```

Return Format 返回格式(同步返回)

application/json,text/json

```
{
        "version":"v1.0",
        "merchantId":"20170912210136",
        "agentAmt":"0.03",
        "currencyCode":"CNY",
        "agentDatetime":"20180115103900",
        "accountNo":"6225666655554444",
        "accountName":"张三",
        "accountType":"2",
        "bankName":"招商银行深圳皇岗支行”,
        "bankCode":"cmb",
        "outAgentNo":"agentpay0215019",
        "bankNo":"11111111111",
        "ext1":"",
        "ext2":"",
        “txnId”:”AG205223321222”,
        “txnDate”:”20170809121200”,
        "signMsg":"hUCnOBTu4P1l8d/mh9ay4blKCZPDiwpzhHLf38FmoZGlTm8cfa5Kbpd6ov1lEfpR46mk+SY
sZzBy\r\nhizAhqqmYHq7oBh9e09mPDmhEYwm6qtD0DQyJOagafK3K/X/yNigZoRpoe+EpT5g+AsR3MBk8h
l2\r\nnsrf3GJHPN6/WwKrhG0=",
        "respCode":"10000",
        "respMsg":"SUCCESS"
}
```

application/xml, text/xml

```
      <version>v1.0</version>
      <merchantId>20170912210136</merchantId>
      <agentAmt>0.03</agentAmt>
      <currencyCode>CNY</currencyCode>
      <agentDatetime>20180115103900</agentDatetime>
      <accountNo>6225666655554444</accountNo>
      <accountName>张三</accountName>
      <accountType>2</accountType>
      <bankName>招商银行深圳皇岗支行</bankName>
      <bankCode>cmb</bankCode>
      <outAgentNo>agentpay0215019</outAgentNo>
      <bankNo>11111111111</bankNo>
      <ext1>ext1</ext1>
      <ext2>ext2</ext2>
      <txnId>AG20160913123212123</txnId>
```

```
        <txnDate>20160913123211</txnDate>
        <respCode>10000</respCode>
        <respMsg>SUCCESS</respMsg>
        <signMsg>hUCnOBTu4P1l8d/mh9ay4blKCZPDiwpzhHLf38FmoZGlTm8cfa5Kbpd6ov1lEfpR46mk+SYsZz
By\r\nhizAhqqmYHq7oBh9e09mPDmhEYwm6qtD0DQyJOagafK3K/X/yNigZoRpoe+EpT5g+AsR3MBk8hl2\
r\nnsrf3GJHPN6/WwKrhG0=</signMsg>
```

下发异步通知

系统处理完成后，会通过该接口通知商户代发结果，通知地址为商户下发时提供的返回地址 agentNotifyUrl,收到 200 表示成功，处理完成返回“success”告知交易系统

参数名称	长度	说明
version	8	与下发接口提交保持一致
merchantId	32	与下发接口提交保持一致
outAgentNo	50	与下发接口提交保持一致
accountNo	32	与下发接口提交保持一致
accountName	32	与下发接口提交保持一致
accountType	1	与下发接口提交保持一致
amt	18	下发金额
bankName	50	与下发接口提交保持一致
bankNo	50	与下发接口提交保持一致
bankCode	20	与下发接口提交保持一致
txnId	32	交易系统返回的下发编号
txnDate	14	交易系统处理完成时间
status	1	5 表示下发成功，其它表示失败
respCode	10	T000 表示处理成功
respMsg	128	错误信息
signMsg	2048	签名信息(验签规则参考 1.2)

Pay attention to the uppercase and lowercase letters of all the fields in interface; **接口中的所有字段，大小写敏感**

2.2.8 汇率查询接口 Exrate Query Interface

Pay attention to uppercase and lowercase letters of all the fields in interface;**接口中的所有字段，大小写敏感**

Address:
Action: POST
Exrate Query Interface 汇率查询接口：
Live 正式地址：https://pgw.alldebitpay.com/rate/queryRate.json

Test 测试地址：http://test-pgw.alldebitpay.com/rate/queryRate.json

Request Format 请求格式

application/json,text/json

Sample:

```
{
        "version":"v1.0",
        "merchantId":"100020160405003",
        " beforeCurrencyCode ":"CNY",
        "afterCurrencyCode ":"USD",
         “merchantSignType”:”1”,
        "signMsg":"ede18ee2fe0171dd0d45be9cc257a54f"
 }
```

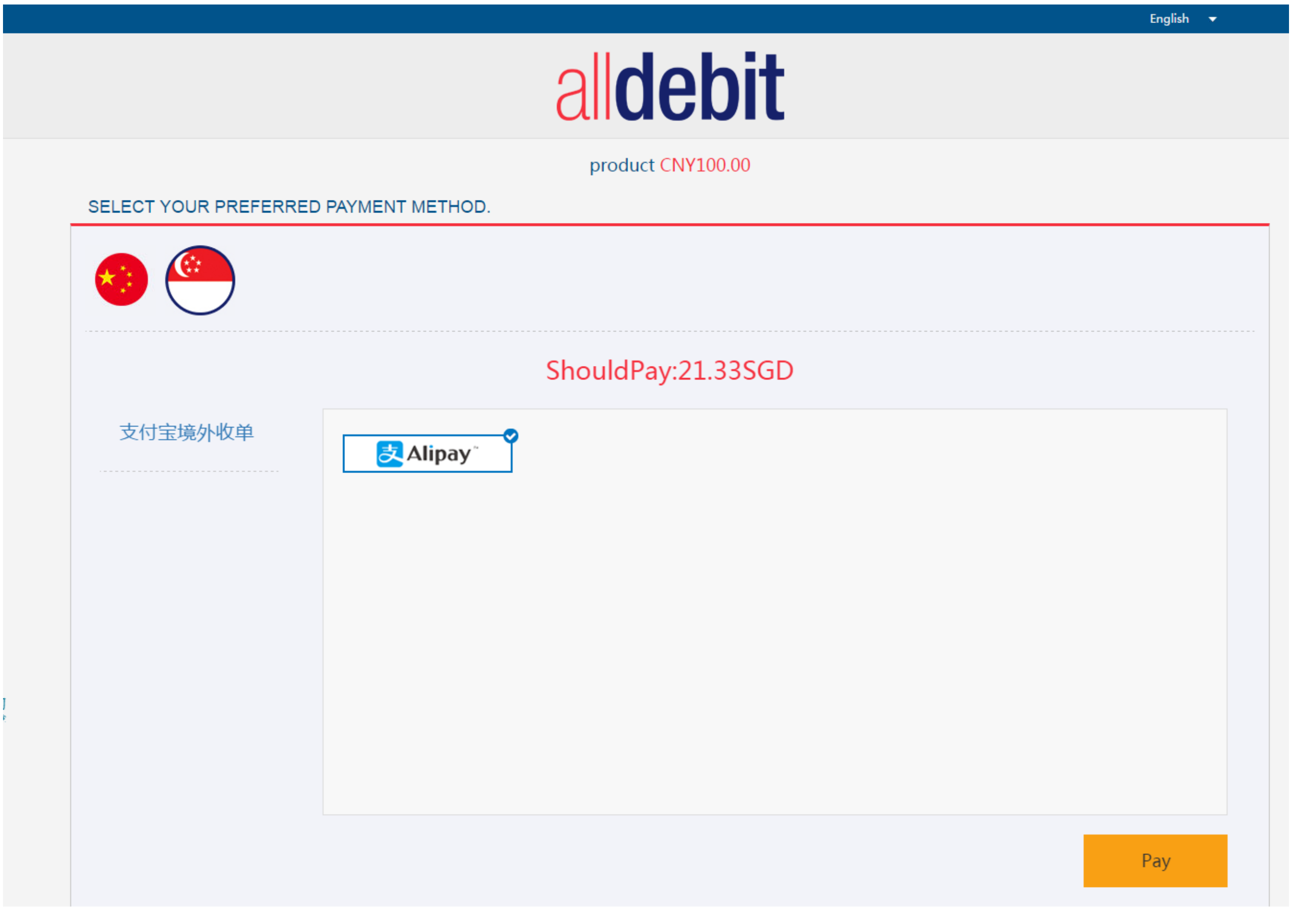
Return Format 返回格式(同步返回)

application/json,text/json

```
{
        " rateList ": [{
                        "afterCurrencyCode":"USD",
                        "expirationDate":"20180806135521",
                        "rate":"0.1467761",
                        "merchantId":"20170926500054",
                        "beforeCurrencyCode":"CNY",
                        "version":"v1.0"
                        }],
          " code ":" T000 ",
        " msg ":"成功",
        "signMsg":"hUCnOBTu4P1l8d/mh9ay4blKCZPDiwpzhHLf38FmoZGlTm8cfa5Kbpd6ov1lEfpR46mk+SYsZzBy\r\nhizAhqqmYHq7oBh9e09mPDmhEYwm6qtD0DQyJOagafK3K/X/yNigZoRpoe+EpT5g+AsR3MBk8hl2\r\nnsrf3GJHPN6/WwKrhG0=",
}
```

3. Test case

Gateway transaction in test environment, please choose 虚拟银行 to complete test; 测试环境的网关交易，通过选择虚拟银行来完成测试。



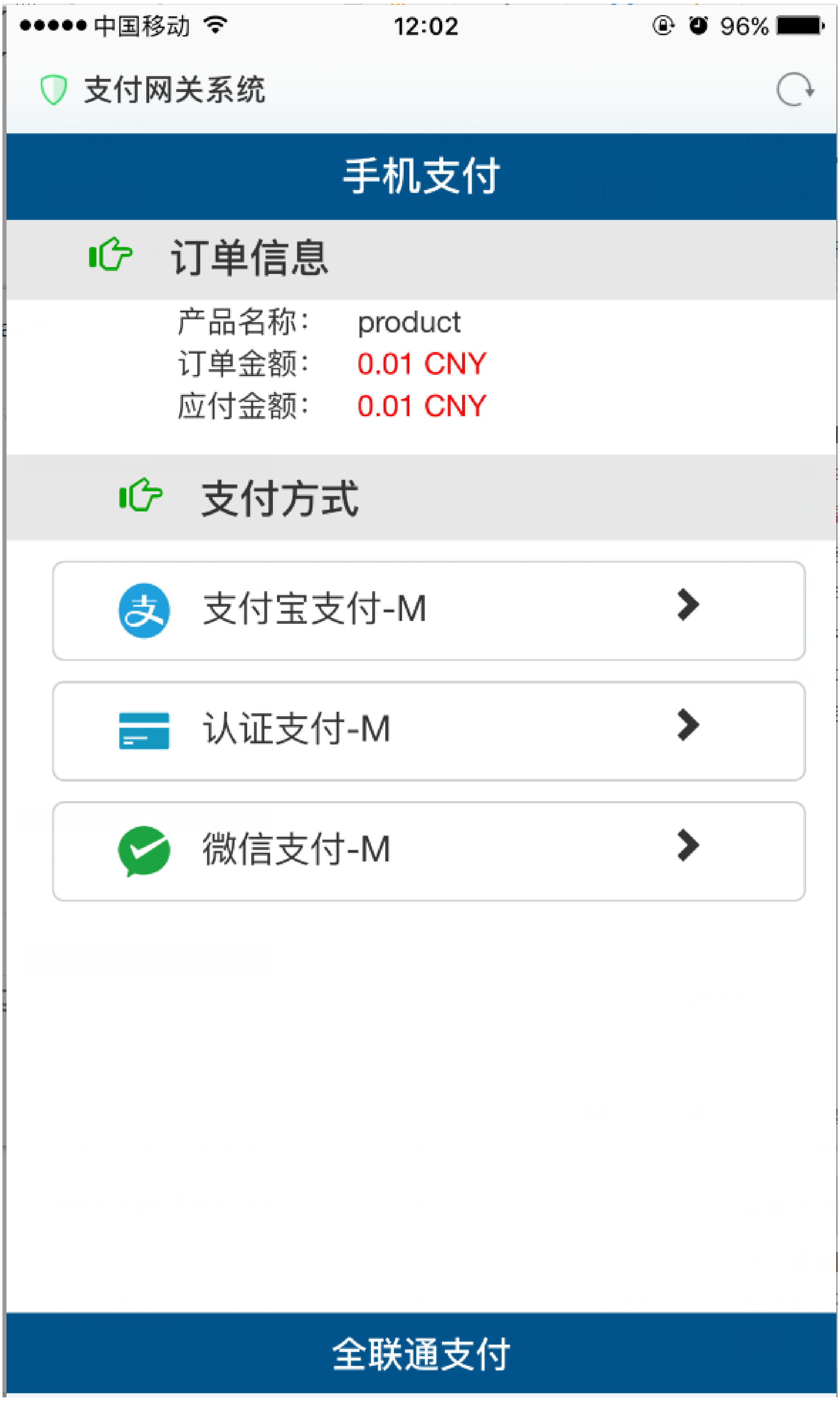
product CNY100.00

SELECT YOUR PREFERRED PAYMENT METHOD.

ShouldPay:21.33SGD

支付宝境外收单

H5 page on Mobile Payment; 移动 H5 支付：



支付网关系统

手机支付

订单信息

产品名称： product
订单金额： 0.01 CNY
应付金额： 0.01 CNY

支付方式

4. 接口定义 Interface definition

4.1 商户注册请求报文描述 Message description of business registration request

字段名	是否必填	长度	描述
version	Y	Fixed Value 固定值	Fixed Value: v 1.0 固定值v1.0
inputCharset	Y	1	1 represents UTF-8; 2 represents GBK; 3 represents GB2312; 1代表UTF-8、2代表GBK、3代表GB2312；
language	Y	1	1 represents simplified Chinese; 2 represents traditional Chinese; 3 represents English; 1代表简体中文、2代表繁体中文、3代表英文
belongOrganize	Y	32	Institutions to which merchants belong 商户所属机构
merchantNameCN	Y	50	Chinese Name of merchant 商户中文名
merchantNameEN	Y	50	English Name of merchant 商户英文名
merchantIndustry	N	50	The industry in which the merchants are located； 商户所在行业
merchantType	Y	1	1：Ordinary merchant 2：Agent
countryType	Y	1	1:china 2:abroad
companyWeb	N	100	Company website;公司网址
companyAddress	Y	100	Company address;公司地址
businessScope	N	100	Scope of business;业务范围
aliasName	N	50	Merchant alias ; 商户别名
contacts	Y	50	contacts；联系人
email	Y	100	email
telephone	Y	50	
zipcode	N	32	Zip code邮编
faxNumber	N	32	
corporateName	Y	50	Name of legal person；法人姓名
corporateIdNumber	Y	50	Corporate Identity Card Number；法人身份证号码

idType	Y	2	ID Type 证件类型 A:Army no B:Business Reg No I:ID Card No N:New IC O:Old IC P:Police no T:Passport
bankCode	Y	32	Receiving Bank Code 收款银行编码
branchCode	Y	32	Receiving Branch Bank Code 收款银行支行编码
accountNo	Y	32	Bank Account No 银行卡号
acName	Y	32	Ac Name
transactionCode	Y	10	Bank Card Type；银行卡类型 20：Misc. Credit 21：Standing Order Credit 22：Salary Credit 23：Dividend Credit 24：Remittance Credit 25：Bill Credit 30：Direct Debit
beneficiaryID	N	32	Beneficiary ID
beneficiaryName1	Y	100	Beneficiary Name 1
beneficiaryName2	N	100	Beneficiary Name 2
beneficiaryName3	N	100	Beneficiary Name 3
beneficiaryName4	N	100	Beneficiary Name 4
beneficiaryAddress1	Y	100	Beneficiary Address 1
beneficiaryAddress2	N	100	Beneficiary Address 2

beneficiaryAddress3	N	100	Beneficiary Address 3
beneficiaryAddress4	N	100	Beneficiary Address 4
beneficiaryCity	N	32	Beneficiary City
beneficiaryPostalCode	N	32	Beneficiary PostalCode
beneficiaryEmail	Y	100	Beneficiary Email
countryCode	Y	3	3-bit country code：china(CHN)；3位国家编码：中国(CHN)
userName	Y	50	
userEmail	Y	50	
phoneNum	Y	50	
signMsg	Y	128	Merchant sign, use RSA to abstract signature. The content of signature is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 商户签名，采用MD5的方式进行签名摘要。签名的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。

4.2 商户注册返回报文描述 Message description of business registration response

字段名	是否必填	长度	描述
Filed Name	Not null	Length	Decription
version	Y	Fixed value 固定值	Fixed Value: v1.0 固定值v1.0
inputCharset	Y	1	1 represents UTF-8; 2 represents GBK; 3 represents

			GB2312;
			1代表UTF-8、2代表GBK、3代表GB2312;

language	Y	1	1 represents simplified Chinese; 2 represents traditional Chinese; 3 represents English; 1代表简体中文、2代表繁体中文、3代表英文
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号
belongOrganize	Y	30	Institutions to which merchants belong 商户所属机构
merchantType	Y	1	1：Ordinary merchant 2：Agent
loginName	Y	50	Login Name
loginPassword	Y	50	Login Password
tradePassword	Y	50	Trade Password
respCode	Y	4	Response code. T000 represents success, refer to the 4.11 of other response code; 响应码。T000表示成功，其余见响应码列表
respMsg	Y	-	Response message; 响应信息
sign	Y	32	Merchant use RSA to verify signature, the content is as below: 1. remove the blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 商户验签，采用MD5的方式进行验签。验签的内容如下: 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。
注意:			

4.3 网银收单请求报文描述 Message description of online banking acquiring request

字段名 Field	是否必填 Not null	长度 Length	描述 Description

version	Y	Fixed Value 固定值	Fixed value: v1.0 固定值v1.0
inputCharset	Y	1	1 represents UTF-8; 2 represents GBK; 3 represents GB2312; 1代表UTF-8、2代表GBK、3代表GB2312；
language	Y	1	1 represents simplified Chinese; 2 represents traditional Chinese; 3 represents English; 1代表简体中文、2代表繁体中文、3代表英文
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号
merOrderNo	Y	50	Character String, only can use letters/numbers/-/_， and shall start from letter or number; No matter succeed or fail, All the order number sent by this merchant , shall be unique. 商户上送的商户订单号，并且只能由数字，字母，”-”，”_”组成，并且以字母或者数字开头 商户订单号不论交易成功或失败，不允许重复
merorderDatetime	Y	14	Order Date time, Date format: yyyyMMddHHmmss, e.g. 20121116020101 订单时间, 日期格式：yyyyMMddHHmmss，例如：20121116020101
merorderCurrency	Y	3	Order Currency, ISO code in 3 digits, ONLY support the currency that merchant has selected when applying MID; 订单币种，三位ISO代码，只支持开通的币种
merorderAmount	Y	12,2	Order amount, maximum hold 2 digits after the decimal point, greater than 0 订单金额，最多保留小数后2位，大于0
pickupUrl	Y	100	Pick up Url, require the whole path in http:// format， without customized parameters such as ?id=123 浏览器返回地址 需http://格式的完整路径，不能加?id=123这类自定义参数
receiveUrl	Y	100	Url of receiving transaction results in backend; require the whole path in http:// format, without customized parameters such as ?id=123 交易结果后台通知地址 需http://格式的完整路径，不能加?id=123这类自定义参数

payType	Y	2	Pay type that consumers can use to make payment, fixed value; 0 represents all, means to display all the pay type that merchant has; TIPS: EC=ECOMMERCE MPM=MERCHANT PRESENT MODE MNP=MINI PROGRAM PAC=PUBLIC ACCOUNT 50: EC-AP-WEB-SGD 63:EC-ENETS-MPM-SGD 64:EC-WP-MPM-USD 71:EC-AP-MPM-SGD 73:EC-SINO-WEB-SGD 74:EC-MGP-WEB-THB 76:EC-MGP-MPM-THB 78:EC-SINO-WEB-CNY 79:EC-SINO-WEB-USD 80:EC-WP-MNP-SGD 86:EC-WP-PAC-SGD 107:EC-WP-MPM-SGD 129:EC-UM-CC-WEB-MYR 130:EC-UM-DD-WEB-MYR 135:EC-AP-MNP-SGD 136:EC-AP-H5-SGD 139:EC-PNOW-MPM-SGD 152:EC-GPAY-WEB-SGD 159:EC-SHPEE-MPM-SGD 161:EC-UPQR-MPM-SGD 171:EC-A+TNG-MPM-SGD 172:EC-A+DN-WEB-SGD 173:EC-A+GC-MPM-SGD 174:EC-A+HK-MPM-SGD 175:EC-A+TM-MPM-SGD 176:EC-A+KKP-MPM-SGD 192:EC-PNOWD-MPM-SGD 206:EC_A+_H5_SGD Remark: For direct payment, please refer to 4.19

issuerId	Y	8	Payment agency ID, should fill in when use direct payment;Refer to 4.12 direct payment and 5.1 institution code. 机构代码，需要直连支付时填写。 参考附录4.19直连支付和5.1支付机构代码
ext1	N	128	Remark 1 备注信息1，
ext2	N	128	Remark 2 备注信息2，
ext3	N	128	Remark 3 备注信息3，
payerName	N	32	Payer’s name, can be Chinese characters or letters; 付款人姓名，可以是汉字，字母
payerEmail	N	50	Payer’s email address, must be email address format; 付款人邮箱，必须是邮箱格式
payerTelephone	N	16	Payer’s phone number, must be mobile phone number; 付款人电话，必须是手机格式
productId	N	20	Product ID; 商品ID
productName	N	256	Product name, can be combined by Chinese characters, letters ,-. 商品名称，可以是汉字，字母，中划线 组成。
productNum	N	4	Product quantity; 商户数量
productPrice	N	20	The unit price of products; 商品单价
productCurrency	N	3	The currency of price, such as CNY, USD; 商品标价币种,如 CNY,USD
businessType	Y	1	Business type, 2 means cross-border business; 业务类型。2：表示跨境业务
merchantSignType	Y	1	Merchant’s Encryption Certificate type: 1 means merchant use RSA key which generate by platform to sign; 2 means merchant use RSA private key which is not generated by platform to sign; Remark: when fill 2, merchant need to provide the RSA private key for the payment system to verify. 3 MD5 Encryption，MD5 key string provided by platform

			商户加密证书方式: 1：表示使用平台生成的商户RSA私钥签名，默认是1 2：表示使用非平台生成的商户RSA私钥签名。 注： 当填写2的时候商户需要给支付系统提供RSA公钥验签 3:MD5加密，由平台提供密钥字符串
signMsg	Y	128	A.Use RSA to abstract signature. The content of signature is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort(not include filed name), and connect the character strings to get the character string after ascending; 3. when merchantSignType=1, use the ciphertext after sign by the RSA private key which generated by platform; when merchantSignType=2, use the ciphertext after sign by the RSA private key which is not generated by platform; B.Use MD5 to abstract signature.The content of signature is as below: 1.Remove blanks and null, sort by field name in ascending; 2.collect the value of the field after sort(not include filed name), then, add “&MD5key” to get the final string. such as, “v1.020140956000.10CNY&zDNuO9vA62qtSMup9jeeOYoRof4fuFfajLQsF6wPmjce7kqlhatmUoqcUBcNPlbubSUWe8TK6tUJGqId42cx2FeOCGDT6U3A80cJvaEUblwtRSVl1MI9FWsVWx50J7Ns” 3.Encrypt with MD5 and convert to uppercase to get signature A.采用RSA的方式进行签名摘要。签名的内容如下: 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 当merchantSignType=1的情况下使用平台生成的商户RSA私钥签名后得到密文，当merchantSignType=2的情况下使用非平台生成的商户RSA私钥签名后得到密文。 B.采用MD5的方式进行签名摘要。签名的内容如下: 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行拼接，末尾加上”&MD5key”得到最终的签名明文。例如:v1.020140956000.10CNY&zDNuO9vA62qtSMup9jeeOYoRof4fuFfajLQsF6wPmjce7kqlhatmUoqcUBcNPlbubSUWe8TK6tUJGqId42cx2FeOCGDT6U3A80c

			JvaEUblwtRSVl1MI9FWsVWx50J7Ns” 3.用MD5加密明文，并转换为大写得到签名结果

4.4 网银收单返回报文描述 Return Message description of online banking acquiring

字段名 Field Name	是否必填 Not Null	长度 Length	描述 Description
version	Y	Fixed Value 固定值	Fixed Value : 1.10 固定值1.1.0
inputCharset	Y	1	1 represents UTF-8; 2 represents GBK; 3 represents GB2312; 1代表UTF-8、2代表GBK、3代表GB2312；
language	Y	1	1 represents simplified Chinese; 2 represents traditional Chinese; 3 represents English; 1代表简体中文、2代表繁体中文、3代表英文
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号
merOrderNo	Y	50	Character String, only can use letters/numbers/-/_， and shall start from letter or number; No matter succeed or fail, all the order number sent by this merchant , shall be unique. 商户上送的商户订单号，并且只能由数字，字母，”-”，”_” 组成，并且以字母或者数字开头 商户订单号不论交易成功或失败，不允许重复
merorderDatetime	Y	14	MID which given by ALLDEBIT; 开放给商户的唯一编号
merorderCurrency	Y	3	Order Currency, ISO code in 3 digits, ONLY support the currency that merchant has selected when applying MID; 订单币种，三位ISO代码，只支持开通的币种
merorderAmount	Y	12,2	Order amount, maximum hold 2 digits after the decimal point, greater than 0 订单金额，最多保留小数后2位，大于0
payAmount	Y	12,2	Payment Amount, maximum hold 2 digits after the decimal point. 付款金额，最多保留小数后2位
payCurrency	Y	3	PaymentCurrency, ISO code in 3 digits

			付款币种，三位ISO代码
payExchangeRate	Y	12,5	The exchange rate when pay, maximum hold7digits after decimal point; 付款时的汇率，最多保留小数后7位
merSTOrderAmount	Y	12,2	Settlement amount, maximum hold 2 digits after decimal point; 结算金额，最多保留小数后2位
merSTCurrency	Y	3	Settlement Currency; 结算币种
exchangeRate	Y	12,5	结算时的汇率，最多保留小数后7位
ext1	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回
ext2	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回
ext3	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回
txnid	Y	20	Transaction serial numbers of ALLDEBIT system; ALLDEBIT的系统流水号
txndate	Y	14	AD System dealing time: yyyMMddHHmmss ALLDEBIT的系统处理时间yyyyMMddHHmmss
status	Y	1	Order status 1. pending; 2. Failed; 3 Success; 订单状态 1.交易中2 支付失败 3：支付成功
tradeFee	Y	5	MDR, 交易手续费
tradeFeeCurrency	Y	3	The currency of MDR; 交易手续费币种
respcode	Y	4	Response code, T000 represents successful; refer to 4.11 for other response code; 响应码。T000表示成功，其余见响应码列表
respmsg	Y	-	Response message; 响应信息
signMsg	Y	32	A.Merchant use RSA to verify signature, the content is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character

			strings to get the character string after ascending; 3. use RSA public key of platform to verify signature, True means verify successfully. B.Merchant use MD5 to verify signature, the content is as below: 1.Remove blanks and null, sort by field name in ascending; 2.collect the value of the field after sort(not include filed name), then, add “&MD5key” to get the final string 3.Encrypt with MD5 and convert to uppercase to get signature A.商户采用RSA的方式进行验签。验签的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 使用平台RSA公钥验签，True表示验证通过。 B.采用MD5的方式进行签名摘要。签名的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行拼接，末尾加上”&MD5key”得到最终的签名明文。例如:v1.020140956000.10CNY&zDNuO9vA62qtSMup9jeeOYoRof4fuFfajLQsF6wPmjce7kqlhatmUoqcUBcNPlbubSUWe8TK6tUJGqId42cx2FeOCGDT6U3A80cJvaEUblwtRSVl1MI9FWsVWx50J7Ns” 3.用MD5加密明文，并转换为大写得到签名结果

4.5 退款请求报文描述 Message description of refund request

字段名	是否必填	长度	描述
version	Y	Fixed Value 固定值	Fixed Value: v 1.0 固定值v1.0

inputCharset	Y	1	1 represents UTF-8; 2 represents GBK; 3 represents GB2312;
			1代表UTF-8、2代表GBK、3代表GB2312;

language	Y	1	1 represents simplified Chinese; 2 represents traditional Chinese; 3 represents English; 1代表简体中文、2代表繁体中文、3代表英文
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号
merOrderNo	Y	50	Merchant’s order number which submitted in original transaction, can be used to confirm the

			refund order; 原交易时上送的商户订单号，用来确认需退款订单
sysOrderNo	Y	20	The unique order number which returned by system when the original transactionsucceed, can be used to confirm the refund order; 原交易成功时系统返回的订单唯一流水号，用来确认需退款订单
refundAmount	Y	12，2	Refund amount; 退款金额
merchantSignType	Y	1	Merchant’s Encryption Certificate type: 1 means merchant uses RSA certificate which generate by platform by default; 2 means merchant uses their own RSA certificate to encrypt; Remark: when fill 2, merchant need to provide payment system RSA certificate to encrypt. 3 MD5 Encryption，MD5 key string provided by platform 商户加密证书方式， 1：表示使用平台默认提供的RSA证书； 2：表示商户需要使用自己的RSA证书加密。 注：当填写2的时候商户需要给支付系统提供RSA证书。 3.MD5加密，由平台提供密钥字符串
refundOrdertime	Y	16	The time when merchant apply for refund, format: yyyyMMddHHmmss 商户退款申请的时间，格式yyyyMMddHHmmss
ext1	N	100	Remark 1 ; 备注字段1
signMsg	Y	128	A.Merchant sign, use RSA to abstract signature. The content of signature is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 3. when merchantSignType=1, use the ciphertext after sign by the RSA private key which generated by platform; when merchantSignType=2, use the ciphertext after sign by the RSA private key which is not generated by platform; B.Merchant use MD5 to verify signature, the content is as below:

			1.Remove blanks and null, sort by field name in ascending; 2.collect the value of the field after sort(not include filed name), then, add “&MD5key” to get the final string 3.Encrypt with MD5 and convert to uppercase to get signature A.商户签名，采用RSA的方式进行签名摘要。签名的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 当merchantSignType=1的情况下使用平台生成的商户RSA私钥签名后得到密文，当merchantSignType=2的情况下使用非平台生成的商户RSA私钥签名后得到密文。 B.采用MD5的方式进行签名摘要。签名的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行拼接，末尾加上”&MD5key”得到最终的签名明文。例如:v1.020140956000.10CNY&zDNuO9vA62qtSMup9jeeOYoRof4fuFfajLQsF6wPmjce7kqlhatmUoqcUBcNPlbubSUWe8TK6tUJGqId42cx2FeOCGDT6U3A80cJvaEUblwtRSVl1MI9FWsVWx50J7Ns” 3.用MD5加密明文，并转换为大写得到签名结果

4.6 退款响应报文描述 Message description of refund response

字段名 Filed Name	是否必填 Not null	长度 Length	描述 Decription
version	Y	Fixed value 固定值	Fixed Value: v1.0 固定值v1.0
inputCharset	Y	1	1 represents UTF-8; 2 represents GBK; 3 represents GB2312; 1代表UTF-8、2代表GBK、3代表GB2312；
language	Y	1	1 represents simplified Chinese; 2 represents traditional Chinese; 3 represents English; 1代表简体中文、2代表繁体中文、3代表英文
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号
merOrderNo	Y	30	The order number which submitted by merchant when applying refund; 退款上送时的商户订单号
refundAmount	Y	12,2	Refund amount; 本次退款金额

refundOrdertime	Y	16	Merchant’s refund application time, format: yyyyMMddHHmmss 商户退款申请的时间，格式yyyyMMddHHmmss
ext1	N	128	Merchant’s order number in system, will return as original when transactions return; 商户系统订单号，在交易返回时会原样返回
status	Y	1	Refund status, 1 represents success, 2 represents fail 退款状态 1-退款成功 2-退款失败
txnId	Y	20	The unique transaction numbers of the refund order which returned by ALLDEBIT; ALLDEBIT返回的该笔退款订单的唯一流水号
txnDate	Y	16	The actual processing time of the refund which returned by ALLDEBIT; ALLDEBIT返回的退款的实际处理时间
respCode	Y	4	Response code. T000 represents success, refer to the 4.11 of other response code; 响应码。T000表示成功，其余见响应码列表
respMsg	Y	-	Response message; 响应信息
signMsg	Y	32	A.Merchant use RSA to verify signature, the content is as below: 1. remove the blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 3. use RSA public key of platform to verify signature, True means verify successfully. B.Merchant use MD5 to verify signature, the content is as below: 1.Remove blanks and null, sort by field name in ascending; 2.collect the value of the field after sort(not include filed name), then, add “&MD5key” to get the final string 3.Encrypt with MD5 and convert to uppercase to get signature 商户验签，采用RSA的方式进行验签。验签的内容如下：

注意：			1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 使用平台RSA公钥验签，True表示验证通过。 B.采用MD5的方式进行签名摘要。签名的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行拼接，末尾加上”&MD5key”得到最终的签名明文。例如:v1.020140956000.10CNY&zDNuO9vA62qtSMup9jeeOYoRof4fuFfajLQsF6wPmjce7kqlhatmUoqcUBcNPlbubSUWe8TK6tUJGqId42cx2FeOCGDT6U3A80cJvaEUblwtRSVl1MI9FWsVWx50J7Ns” 3.用MD5加密明文，并转换为大写得到签名结果

4.7 订单查询请求报文描述 Message description of order querying request

字段名 Filed Name	是否必填 Not null	长度 Length	描述 Description
version	Y	Fixed Value 固定值	Fixed Value: v1.0 固定值v1.0
merchantId	Y	16	MID which given by ALLDEBIT 开放给商户的唯一编号
merOrderNo	Y	50	Order number which submitted by merchant when acquiring; 收单时商户上送的商户订单号
merorderDatetime	Y	14	Order time which submitted by merchant, format: yyyyMMddHHmmss 收单时商户上送的订单时间，格式yyyyMMddHHmmss
merchantSignType	Y	1	Merchant’s Encryption Certificate type: 1 means merchant uses RSA certificate which generate by platform by default; 2 means merchant uses their own RSA certificate to encrypt; Remark: when fill 2, merchant need to provide payment system RSA certificate to encrypt. 3 MD5 Encryption，MD5 key string provided by platform 商户加密证书方式， 1：表示使用平台默认提供的RSA证书； 2：表示商户需要使用自已的RSA证书加密。 注：当填写2的时候商户需要给支付系统提供RSA证书。 3.MD5加密，由平台提供密钥字符串
signMsg	Y	128	A.Merchant use RSA to verify signature, the content

			is as below: 1. remove the blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 3. use RSA public key of platform to verify signature, True means verify successfully. B.Merchant use MD5 to verify signature, the content is as below: 1.Remove blanks and null, sort by field name in ascending; 2.collect the value of the field after sort(not include filed name), then, add “&MD5key” to get the final string 3.Encrypt with MD5 and convert to uppercase to get signature 商户验签，采用RSA的方式进行验签。验签的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 使用平台RSA公钥验签，True表示验证通过。 B.采用MD5的方式进行签名摘要。签名的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行拼接，末尾加上”&MD5key”得到最终的签名明文。例如:v1.020140956000.10CNY&zDNuO9vA62qtSMup9jeeOYoRof4fuFfajLQsF6wPmjce7kqlhatmUoqcUBcNPlbubSUWe8TK6tUJGqId42cx2FeOCGDT6U3A80cJvaEUblwtRSVl1MI9FWsVWx50J7Ns” 3.用MD5加密明文，并转换为大写得到签名结果

4.8 订单查询响应报文描述 Response message description of Order query

字段名	是否必填	长度	描述
version	Y	固定值	Fixed value: v1.0 固定值v1.0
inputCharset	Y	1	1 represents UTF-8; 2 represents GBK; 3 represents GB2312; 1代表UTF-8、2代表GBK、3代表GB2312；
language	Y	1	1 represents simplified Chinese; 2 represents traditional Chinese; 3 represents English;

			1代表简体中文、2代表繁体中文、3代表英文
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号
merOrderNo	Y	50	Character String, only can use letters/numbers/-/_， and shall start from letter or number; No matter succeed or fail, All the order number sent by this merchant , shall be unique. 商户上送的商户订单号，并且只能由数字，字母，”-”,”_”组成，并且以字母或者数字开头 商户订单号不论交易成功或失败，不允许重复
merorderDatetime	Y	14	Order time, Date format: yyyyMMddHHmmss, e.g.:20121116020101 订单时间 日期格式：yyyyMMddHHmmss，例如：20121116020101
merorderCurrency	Y	3	Order Currency, ISO code in 3 digits, 订单币种，三位ISO代码
merorderAmount	Y	12,2	Order amount, maximum hold 2 digits after the decimal point 订单金额，最多保留小数后2位
ext1	N	128	Remark, will return as original when transactions return; 备注信息，在交易返回时会原样返回
ext2	N	128	Remark, will return as original when transactions return; 备注信息，在交易返回时会原样返回
ext3	N	128	Remark, will return as original when transactions return; 备注信息，在交易返回时会原样返回
txnId	Y	20	Transaction serials numbers of ALLDEBIT; ALLDEBIT的系统流水号
txnDate	Y	14	AD system processing time: yyyyMMddHHmmss ALLDEBIT的系统处理时间yyyyMMddHHmmss
state	Y	1	Order Status, 1. Pending; 2. Failed, 3, success 订单状态 1.交易中 2 支付失败 3：支付成功
respCode	Y	4	Response code, T000 represents success, refer to the 4.11 for other response code; 响应码。T000表示成功，其余见响应码列表
respMsg	Y	-	Response message; 响应信息

字段名	是否必填	长度	描述
signMsg	Y	32	A.Merchant use RSA to verify signature, the content is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 3. use RSA public key of platform to verify signature, True means verify successfully. B.Merchant use MD5 to verify signature, the content is as below: 1.Remove blanks and null, sort by field name in ascending; 2.collect the value of the field after sort(not include filed name), then, add “&MD5key” to get the final string 3.Encrypt with MD5 and convert to uppercase to get signature A.采用RSA的方式进行验签。验签的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 使用平台RSA公钥验签，True表示验证通过。 B.采用MD5的方式进行签名摘要。签名的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行拼接，末尾加上”&MD5key”得到最终的签名明文。例如:v1.020140956000.10CNY&zDNuO9vA62qtSMup9jeeOYoRof4fuFfajLQsF6wPmjce7kqlhatmUoqcUBcNPlbubSUWe8TK6tUJGqId42cx2FeOCGDT6U3A80cJvaEUblwtRSVl1MI9FWsVWx50J7Ns” 3.用MD5加密明文，并转换为大写得到签名结果
注意：			

4.9 指定结算请求报文描述 Message description of appointed settlement request

字段名	是否必填	长度	描述
version	Y	Fixed Value 固定值	Fixed value: v1.0 固定值v1.0

inputCharset	Y	1	1 represents UTF-8; 2 represents GBK; 3 represents GB2312; 1代表UTF-8、2代表GBK、3代表GB2312；
language	Y	1	1 represents simplified Chinese; 2 represents traditional Chinese; 3 represents English; 1代表简体中文、2代表繁体中文、3代表英文
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号
refcnceFlow	Y	30	Reference numbers in system; 系统流水号
merchantSignType	Y	1	Merchant’s Encryption Certificate type: 1 means merchant use RSA certificate which provided by platform by default; 2 means merchant use their own RSA certificate to encrypt; Remark: when fill 2, merchant need to provide the RSA certificate to payment system. 商户加密证书方式，1：表示使用平台默认提供的RSA证书，2：表示商户需要使用自己的RSA证书加密。 注： 1.当填写2的时候商户需要给支付系统提供RSA证书
merOrderNo	Y	50	The merchant order number which submitted by merchant when acquiring; 收单时商户上送的商户订单号
shouldDealtime	Y	8	The date when settlement should be made, e.g.: 20170410 应结算日期，例如：20170410
state	Y	1	Settlement type: 1 represents it is settled according to appointed time; 结算类型，1表示指定时间结算
bussinesType	Y	1	Business Type: 1 represents CNY business, 2 represents cross-border business; 业务类型，1表示人民币业务，2跨境业务
signMsg	Y	128	Merchant sign, use RSA to abstract signature. The content of signature is as below: 1. remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 3. when merchantSignType=1, use the ciphertext after sign by the RSA private key which generated by platform; when merchantSignType=2, use the

			ciphertext after sign by the RSA private key which is not generated by platform; 商户签名，采用RSA的方式进行签名摘要。签名的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 当merchantSignType=1的情况下使用平台生成的商户RSA私钥签名后得到密文，当merchantSignType=2的情况下使用非平台生成的商户RSA私钥签名后得到密文。

4.10 指定结算响应报文描述 Response message description of appointed settlement

字段名 Field Name	是否必填 Not null	长度 Length	描述 Description
version	Y	固定值	Fixed value: v1.0 固定值v1.0
inputCharset	Y	1	1 represents UTF-8; 2 represents GBK; 3 represents GB2312; 1代表UTF-8、2代表GBK、3代表GB2312；
language	Y	1	1 represents simplified Chinese; 2 represents traditional Chinese; 3 represents English; 1代表简体中文、2代表繁体中文、3代表英文
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号
merOrderNo	Y	50	The merchant order number which submitted by merchant when acquiring; 收单时商户上送的商户订单号
refcnceFlow	Y	30	Reference numbers in system; 系统流水号
state	Y	1	Settlement type: 1 represents it is settled according to appointed time; 结算类型，1表示指定时间结算
merchantSignType	Y	1	Merchant’s Encryption Certificate type: 1 means merchant use RSA certificate which provided by platform by default; 2 means merchant use their own RSA certificate to encrypt; Remark: when fill 2, merchant need to provide the RSA certificate to payment system. 商户加密证书方式，1：表示使用平台默认提供的RSA证书，2：表示商户需要使用自己的RSA证书加密。 注： 1.当填写2的时候商户需要给支付系统提供RSA证书
shouldDealtime	Y	8	The date when settlement should be made, e.g.:

			20170410 应结算日期，例如：20170410
bussinesType	Y	1	Business Type: 1 represents CNY business, 2 represents cross-border business; 业务类型，1表示人民币业务，2跨境业务
respCode	Y	4	Response code, T000 represents success, refer to the 4.11 for other response code; 响应码。T000表示成功，其余见响应码列表
respMsg	Y	-	Response message; 响应信息
signMsg	Y	32	Merchant use RSA to verify signature, the content is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 3. use RSA public key of platform to verify signature, True means verify successfully. 商户验签，采用RSA的方式进行验签。验签的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 使用平台RSA公钥验签，True表示验证通过。

注意：

4.11 移动支付请求报文描述 Message description of mobile payment request

字段名 Field Name	是否必填 Not null	长度 Length	描述 Description
version	Y	固定值	Fixed value: v1.0 固定值v1.0
inputCharset	Y	1	1 represents UTF-8; 2 represents GBK; 3 represents GB2312; 1代表UTF-8、2代表GBK、3代表GB2312；
language	Y	1	1 represents simplified Chinese; 2 represents traditional Chinese; 3 represents English; 1代表简体中文、2代表繁体中文、3代表英文
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号

merOrderNo	Y	50	Character String, only can use letters/numbers/-/_， and shall start from letter or number; No matter succeed or fail, All the order number sent by this merchant , shall be unique. 商户上送的商户订单号，并且只能由数字，字母，”-”,”_”组成，并且以字母或者数字开头 商户订单号不论交易成功或失败，不允许重复
merorderDatetime	Y	14	Order time, Date format: yyyyMMddHHmmss, e.g.:20121116020101 订单时间 日期格式：yyyyMMddHHmmss，例如：20121116020101
merorderCurrency	Y	3	Order Currency, ISO code in 3 digits, 订单币种，三位ISO代码
merorderAmount	Y	12,2	Order amount, maximum hold 2 digits after the decimal point 订单金额，最多保留小数后2位
pickupUrl	Y	100	Pickup Url Require the whole path in http:// format, without customized parameters such as ?id=123 浏览器返回地址 需http://格式的完整路径，不能加?id=123这类自定义参数
receiveUrl	Y	100	Receiving address of transaction result in backend, Require the whole path in http:// format, without customized parameters such as ?id=123 交易结果后台通知地址 需http://格式的完整路径，不能加?id=123这类自定义参数
payType	Y	2	暂无移动支付产品 There are no mobile payment products for the time being
issuerId	Y	8	Payment agency ID, should fill in when use direct payment;Refer to 4.12 direct payment and 5.1 institution code; 机构代码，需要直连支付时填写。 参考附录4.12直连支付和5.1支付机构代码机
ext1	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回

ext2	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回
ext3	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回
payerAcctNo	N	32	Payer’ s information, spare, nullable; 付款人信息，备用，可空
termId	N	6	Required when use APP payment, can be null when use other payment, shows the type of terminal; 1.IOS , 2.ANDROID , 3.WP(windows phone) APP 支付必填，其他交易不填，表明终端类型： 1.IOS , 2.ANDROID , 3.WP(windows phone) 例如：IOS终端上送：IOS
productName	N	256	Product name, can be combined withChinese characters, letters or - 商品名称，可以是汉字，字母，中划线 组成。
businessType	Y	1	Business type. 1 represents CNY business, 2 represents cross-border business 业务类型。1：表示人民币业务 2跨境业务
merchantSignType	Y	1	Merchant encrypts certificate, 1: means to use the RSA certificate which provided by platform by default; 2 : means Merchant uses their own RSA certificate to encrypt; Remark: when fill 2, means merchant need to provide payment system the RSA certificate. Merchant sign, use RSA to abstract signature. The content of signature is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 3. when merchantSignType=1, use the ciphertext after sign by the RSA private key which generated by platform; when merchantSignType=2, use the ciphertext after sign by the RSA private key which is not generated by platform; 商户加密证书方式，1：表示使用平台默认提供的RSA证书 2：表示商户需要使用自己的RSA证书加密。 注： 1.当填写2的时候商户需要给支付系统提供RSA证书

signMsg	Y	128	商户签名，采用RSA的方式进行签名摘要。签名的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 当merchantSignType=1的情况下使用平台生成的商户RSA私钥签名后得到密文，当merchantSignType=2的情况下使用非平台生成的商户RSA私钥签名后得到密文。

4.12 移动支付响应报文描述 Response Message description of mobile payment

字段名	是否必填	长度	描述
Field Name	**Not null**	**Length**	**Description**
version	Y	固定值	Fixed value: v1.0 固定值v1.0
inputCharset	Y	1	1 represents UTF-8; 2 represents GBK; 3 represents GB2312; 1代表UTF-8、2代表GBK、3代表GB2312；
language	Y	1	1 represents simplified Chinese; 2 represents traditional Chinese; 3 represents English; 1代表简体中文、2代表繁体中文、3代表英文
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号
merOrderNo	Y	50	Character String, only can use letters/numbers/-/_， and shall start from letter or number; No matter succeed or fail, All the order number sent by this merchant , shall be unique. 商户上送的商户订单号，并且只能由数字，字母，”-”，”_”组成，并且以字母或者数字开头 商户订单号不论交易成功或失败，不允许重复
merorderDatetime	Y	14	Order time, Date format: yyyyMMddHHmmss, e.g.:20121116020101 订单时间 日期格式：yyyyMMddHHmmss，例如：20121116020101
merorderCurrency	Y	3	Order Currency, ISO code in 3 digits, 订单币种，三位ISO代码
merorderAmount	Y	12,2	Order amount, maximum hold 2 digits after the decimal point 订单金额，最多保留小数后2位
payAmount	Y	12,2	Payment amount, maximum hold 2 digits after the decimal point 付款金额，最多保留小数后2位
payCurrency	Y	3	Payment currency, ISO code in 3 digits

			付款币种，三位ISO代码
payExchangeRate	Y	12,5	The exchange rate when making payment, maximum hold 7 digits after decimal point; 付款时的汇率，最多保留小数后7位
merSTOrderAmount	Y	12,2	Settlement Amount, maximum hold 2 digits after decimal point; 结算金额，最多保留小数后2位
merSTCurrency	Y	3	Settlement currency; 结算币种
exchangeRate	Y	12,5	The exchange rate when settling, maximum hold 7 digits after decimal point; 结算时的汇率，最多保留小数后7位
ext1	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回
ext2	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回
ext3	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回
txnid	Y	20	Transaction serials numbers in AD system; ALLDEBIT的系统流水号
txndate	Y	14	AD system processing time: yyyyMMddHHmmss ALLDEBIT的系统处理时间yyyyMMddHHmmss
status	Y	1	Order status: 1 .pending; 2. Failed; 3. Success. 订单状态1.未付款 2 支付失败 3：支付成功
tradeFee	Y	5	MDR; 交易手续费
tradeFeeCurrency	Y	3	MDR currency; 交易手续费币种
respcode	Y	4	Response code, T000 represents success, refer to 4.11 for other response code; 响应码。T000表示成功，其余见响应码列表
respmsg	Y	-	Response message; 响应信息
signMsg	Y	32	Merchant use RSA to verify signature, the content is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending;

			3. use RSA public key of platform to verify signature, True means verify successfully. 商户验签，采用RSA的方式进行验签。验签的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 使用平台RSA公钥验签，True表示验证通过。
注意：			

4.13 下发请求报文描述 Message description of agent request

字段名 Field Name	是否必填 Not null	长度 Length	描述 Description
version	Y	固定值	Fixed value: v1.0 固定值v1.0
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号
agentAmt	Y	12,2	Agent amount, maximum hold 2 digits after the decimal point 下发金额，最多保留小数后2位
agentDatetime	Y	14	Agent time, Date format: yyyyMMddHHmmss, e.g.:20121116020101 下发时间 日期格式：yyyyMMddHHmmss，例如：20121116020101
currencyCode	Y	3	AgentCurrency, ISO code in 3 digits, 下发币种，三位ISO代码
accountNo	Y	30	银行账号
accountName	Y	30	开户行姓名
accountType	Y	1	账户类别 1对公 2对私
bankName	Y	50	开户行支行名称
bankCode	Y	30	银行编码
outAgentNo	Y	30	下发的商户订单号，保证唯一
bankNo	Y	30	开户行联行号
ext1	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回
ext2	N	128	Remark, will return as original when transaction

			returns; 备注信息，在交易返回时会原样返回
agentNotifyUrl	Y	256	异步通知地址如:**http://www.test.com/aa.do** 不能带？**Param=123**等格式
merchantSignType	Y	1	Merchant encrypts certificate, 1: means to use the RSA certificate which provided by platform by default; 2 : means Merchant uses their own RSA certificate to encrypt; Remark: when fill 2, means merchant need to provide payment system the RSA certificate. Merchant sign, use RSA to abstract signature. The content of signature is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 3. when merchantSignType=1, use the ciphertext after sign by the RSA private key which generated by platform; when merchantSignType=2, use the ciphertext after sign by the RSA private key which is not generated by platform; 商户加密证书方式，1：表示使用平台默认提供的RSA证书 2：表示商户需要使用自己的RSA证书加密。 注： 1.当填写2的时候商户需要给支付系统提供RSA证书
signMsg	Y	256	商户签名，采用RSA的方式进行签名摘要。签名的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 当merchantSignType=1的情况下使用平台生成的商户RSA私钥签名后得到密文，当merchantSignType=2的情况下使用非平台生成的商户RSA私钥签名后得到密文。

4.14 下发请求同步响应报文描述 Response Message description of agent

字段名 **Field Name**	**是否必填** **Not null**	长度 **Length**	**描述** **Description**
version	Y	固定值	Fixed value: v1.0 固定值v1.0
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号

agentAmt	Y	12,2	Agent amount, maximum hold 2 digits after the decimal point 下发金额，最多保留小数后2位
agentDatetime	Y	14	Agent time, Date format: yyyyMMddHHmmss, e.g.:20121116020101 下发时间 日期格式：yyyyMMddHHmmss，例如：20121116020101
currencyCode	Y	3	Agent Currency, ISO code in 3 digits, 下发币种，三位ISO代码
accountNo	Y	30	银行卡号
accountName	Y	30	开户行姓名
accountType	Y	1	账户类别 1对公 2对私
bankName	Y	50	开户行支行名称
bankCode	Y	30	银行编码
outAgentNo	Y	30	下发的商户订单号，保证唯一
bankNo	Y	30	开户行联行号
ext1	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回
ext2	N	128	Remark, will return as original when transaction returns; 备注信息，在交易返回时会原样返回
txnId	Y	20	Transaction serials numbers in AD system; ALLDEBIT的系统流水号
txnDate	Y	14	AD system processing time: yyyyMMddHHmmss ALLDEBIT的系统处理时间yyyyMMddHHmmss
respCode	Y	4	Response code, 10000 represents success, refer to 4.11 for other response code; 响应码。10000表示成功，其余见响应码列表
respMsg	Y	-	Response message; 响应信息
signMsg	Y	32	Merchant use RSA to verify signature, the content is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending; 3. use RSA public key of platform to verify signature, True means verify successfully. 商户验签，采用RSA的方式进行验签。验签的内容如下：

1.去掉空串和null后，按字段名升序排序。
2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。
3. 使用平台RSA公钥验签，True表示验证通过。

注意：

4.15 下发请求异步返回字段描述

字段名 Field Name	是否必填 Not null	长度 Length	描述 Description
version	Y	固定值	Fixed value: v1.0 固定值v1.0
merchantId	Y	16	MID which given by ALLDEBIT; 开放给商户的唯一编号
outAgentNo	Y	50	Character String, only can use letters/numbers/-/_， and shall start from letter or number; No matter succeed or fail, All the order number sent by this merchant , shall be unique. 商户上送的商户下发订单号，并且只能由数字，字母，”-”,”_”组成，并且以字母或者数字开头 商户下发订单号不论交易成功或失败，不允许重复
accountNo	Y	30	银行卡号
accountName	Y	3	开户人姓名
accountType	Y	1	账户类型 1对公 2对私
amt	Y	12,2	Agent amount, maximum hold 2 digits after the decimal point 下发金额，最多保留小数后2位
bankName	Y	50	银行支行名称
bankNo	Y	30	银行联行号
bankCode	Y	30	银行编码如招商银行：cmb
txnId	Y	20	Transaction serials numbers in AD system; ALLDEBIT的系统流水号
txnDate	Y	14	AD system processing time: yyyyMMddHHmmss ALLDEBIT的系统处理时间yyyyMMddHHmmss
status	Y	1	5表示下发成功，其它表示失败
respCode	Y	4	Response code, T000 represents success, refer to 4.11 for other response code;

			响应码。T000表示成功，其余见响应码列表
respMsg	Y	-	Response message; 响应信息
signMsg	Y	32	Merchant use RSA to verify signature, the content is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending; 3. use RSA public key of platform to verify signature, True means verify successfully. 商户验签，采用RSA的方式进行验签。验签的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 使用平台RSA公钥验签，True表示验证通过。

注意：

4.16 汇率查询请求报文描述 Message description of exrate querying request

字段名 Filed Name	是否必填 Not null	长度 Length	描述 Description
version	Y	Fixed Value 固定值	Fixed Value: v1.0 固定值v1.0
merchantId	Y	16	MID which given by ALLDEBIT 开放给商户的唯一编号
beforeCurrencyCode	Y	3	Old Currency, ISO code in 3 digits, 原币种，三位ISO代码
afterCurrencyCode	Y	3	Target Currency, ISO code in 3 digits, 目标币种，三位ISO代码
merchantSignType	Y	1	Merchant encrypts certificate, 1: means to use the RSA certificate which provided by platform by default; 2 : means Merchant use their own RSA certificate to encrypt; Remark: when fill 2, means merchant need to provide payment system the RSA certificate. Merchant sign, use RSA to abstract signature. The content of signature is as below:

			1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 3. when merchantSignType=1, use the ciphertext after sign by the RSA private key which generated by platform; when merchantSignType=2, use the ciphertext after sign by the RSA private key which is not generated by platform; 商户加密证书方式，1：表示使用平台默认提供的RSA证书 2：表示商户需要使用自己的RSA证书加密。 注： 1.当填写2的时候商户需要给支付系统提供RSA证书
signMsg	Y	128	商户签名，采用RSA的方式进行签名摘要。签名的内容如下： 1.去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。 3. 当merchantSignType=1的情况下使用平台生成的商户RSA私钥签名后得到密文，当merchantSignType=2的情况下使用非平台生成的商户RSA私钥签名后得到密文。

4.17 汇率查询响应报文描述 Response message description of Exrate query

字段名	是否必填	长度	描述
code	Y	4	Response code, T000 represents success, refer to the 4.11 for other response code; 响应码。T000表示成功，其余见响应码列表固定值v1.0
msg	Y	1	Response message; 响应信息
rateList	Y	5000	返回汇率列表
signMsg	Y	128	Merchant use RSA to verify signature, the content is as below: 1. Remove blanks and null, sort by field name in ascending; 2. collect the value of the field after sort (not include field name), and connect the character strings to get the character string after ascending (remove null and blanks); 3. use RSA public key of platform to verify signature, True means verify successfully. 商户验签，采用RSA的方式进行验签。验签的内容如下： 1. 去掉空串和null后，按字段名升序排序。 2. 取排序后字段的值（不包含字段名）进行字符串拼接得到字符串。

3. 使用平台RSA公钥验签，True表示验证通过。

注意:

4.18 响应码/消息列表 Response message list

Respcode	Response message	
T000	成功	Successful
T001	系统错误	System error
T002	请求超时	Request timeout
T003	商户订单号重复	Merchant order NO. repeated
PGW000	成功	Successful
PGW001	失败	Failed
PGW003	请求超时	Request timeout
GW0001	请求报文为空	Request message is blank
GW0002	版本号不正确	Incorrect version No.
GW0003	字符集不正确	Incorrect character string
GW0004	语言不正确	Incorrect Language
GW0005	商户号不正确	Incorrect MID
GW0006	商户密钥不正确	Incorrect Merchant KEY
GW0007	订单号不合法	Invalid order No.
GW0008	订单时间不正确	Incorrect order time
GW0009	订单币种不正确	Incorrect order currency
GW0010	不支持的订单币种	Order currency is not supported
GW0011	数据查询异常	Abnormal data query
GW0012	浏览器返回地址不正确	Incorrect pickup Url of browser
GW0013	服务器返回地址不正确	Incorrect pickup Url of server
GW0014	未注册的URL	Unregistered Url
GW0015	URL地址不正确	Incorrect Url
GW0016	备注1不正确	Incorrect Remark 1

GW0017	备注2不正确	Incorrect Remark 2
GW0018	备注3不正确	Incorrect Remark 3
GW0019	付款人名称长度不正确	Incorrect length of payer’s name
GW0020	付款人邮箱不正确	Incorrect payer’s email
GW0021	付款人电话不正确	Incorrect payer’s phone number
GW0022	产品ID不合法	Invalid product ID
GW0023	产品名称不合法	Invalid product name
GW0024	产品数量不合法	Invalid product quantity
GW0025	产品价格不正确	Invalid product price
GW0026	未配置手续费	MDR unset
GW0027	查询不到可用的交易通道	Can not query available transaction channels
GW0028	签名错误	Signature error
GW0029	原订单号不存在	Original order number doesn’t exist
GW0030	必须是支付成功的订单才能退款	Only successful payment order can do refund
GW0031	可用退款订单金额不足	Amount for order refund is not enough.
GW0032	退款金额不能大于账户当前余额	Refund amount can’t be greater than current balance of account
GW0033	订单流水号不正确	Incorrect order No.
GW0034	金额不正确	Incorrect Amount
GW0035	退款时间不正确	Incorrect refund time
GW0036	商户RSA证书不存在	Merchant’s RSA certificate doesn’t exist.
GW0037	商户RSA证书签名不通过	Merchant’s RSA certificate doesn’t pass.
GW0038	接口调用异常	A
GW0039	浏览器地址不合法	Invalid browser url
GW0040	服务器地址不合法	Invalid server url
GW0041	退款字符串不合法	Invalid refund character string
GW0042	请待系统清算完再退款	
GW0043	银行机构代码不合法	Invalid bank issuer ID
GW0044	退款源不合法	Can’t find the payment channel

GW0045	退款商户号不合法	Invalid refund MID
GW0046	原交易不存在	Original transaction doesn’t exist
GW0050	商户配置不正确	Incorrect merchant setting
GW0051	订单未支付成功，不允许退款	Order payment is not successful, not allow to do refund
GW0052	Paytype不合法	Invalid Paytype
GW0053	金额限额超限	Amount runs over limits.
GW0054	没有查询到符合的记录	No records were found in compliance
GW0057	订单金额不足以扣手续费	Order amount did not afford MDR
GW0058	代理商不能发起支付	Agent cannot initiate payment
GW0060	退款金额不能大于实际所得金额	Refund amount shall not be greaterthan
GW0061	提交参数不完整	The parameters submitted is incomplete.
GW0062	结算状态异常	Abnormal settlement status
GW0063	日期(时间)格式异常	Abnormal date(time) format
GW0064	商户账户未配置	Merchant account is not set
GW0065	商户扣率未配置	
GW0066	商户证书未配置	Merchant certificate is not set.
ITS0001	版本号不合法	Invalid version No.
ITS0002	字符集不合法	Invalid character set.
ITS0003	语言不合法	Invalid language
ITS0004	接入机构号不合法	Invalid institution code
ITS0005	接入机构号下商户未注册	This MID can’t be found under this institution code
ITS0006	商户订单号不合法	Invalid merchant order number
ITS0007	订单时间不合法	Invalid order time
ITS0008	币种不合法	Invalid currency
ITS0009	订单金额不合法	Invalid order amount
ITS0010	订单结算金额不合法	Settlement amount of order is invalid.
ITS0011	浏览器返回地址不合法	Invalid pickup url of browser
ITS0012	服务器返回地址不合法	Invalid pickup url of server

ITS0013	支付方式不合法	Invalid Payment method
ITS0014	机构代码不合法	Invalid agent ID
ITS0015	备注1不合法	Invalid Remark 1
ITS0016	备注2不合法	Invalid Remark 2
ITS0017	备注3不合法	Invalid Remark 3
ITS0018	付款人姓名不合法	Invalid payer’s name
ITS0019	付款人邮箱不合法	Invalid payer’s email address
ITS0020	付款人电话不合法	Invalid payer’s phone number
ITS0021	商品ID不合法	Invalid product ID
ITS0022	商品名称不合法	Invalid product name
ITS0023	商品数量不合法	Invalid product quantity
ITS0024	商品单价不合法	Invalid product price
ITS0025	商品标价币种不合法	Invalid product currency
ITS0027	签名不正确	Invalid signature
ITS0028	商户通道不存在	Invalid
ITS0029	代付请求失败	Invalid withdrawal request
ITS0030	退款失败	Refund failed
ITS0031	订单流水号不存在	Order number doesn’t exist.
ITS0032	退款类型不正确	Incorrect refund type
ITS0033	全额退款时退款金额必须和订单金额相等	Refund amount must be equal to order amount when doing full refund.
ITS0034	商户号已存在	MID
ITS0035	商户中文名不能为空	Merchant’s Chinese name can’t be null.
ITS0036	商户英文名不能为空	Merchant’s English name can’t be null.
ITS0037	没有匹配到交易通道	Payment channel doesn’t match.
ITS0038	结算状态异常	Abnormal settlement status
ITS0039	请求报文参数不全	Incomplete request message parameter
ITS0040	没有查询到符合条件的数据	No records were found in compliance
ITS0041	数据查询异常	Abnormal data querying

ITS0042	商户英文名不正确	Incorrect merchant English name
ITS0043	支付成功的订单才能退款	Only successful payment order can do refund
ITS0044	可用退款金额不足	Amount to be refund is not enough.

4.19 直连支付 Direct Payment

If merchants don’twant to display ALLDEBIT’s payment page when consumers doing payment, can use direct payment. Fill payType(not 0) and isserId in the interface to use direct payment.

如果在支付过程中不出现 ALLDEBIT 的网关收银台页面，可以采用直连支付的方式。在接口中同时填写 payType(不为 0)和 issuerId 即为直连支付.

1. Overseas Wechat is scanned directly, fixed payType = 64 or 107 issuerId = wechat, interface returns JSON string synchronously, such as

```
{"respCode":"T000","respMsg":"成功
","code_url":"weixin://wxpay/bizpayurl?pr=uduaLGe","sign":"dpYZPOzdeTbSC1crvwm4EevBU
7X1xU5ee3rAYSFv5hgAPjQxkTaE0bhkBPiuOLMi0Giq8yCD+5fZ\nQHutkr8c93FUo7cecS+T+Ejtq
UjUnnahmw7ZKBmWA+RxYPg2jNsWT+s1zltBADe7/pjSW2h8VYuE\nT+FrGB0I24lY3Ga/rmo="}
```

海外微信扫码直连，固定填写 payType=64 或 107 issuerId=wechat,接口同步返回 json 字符串如：

```
{"respCode":"T000","respMsg":"成功
","code_url":"weixin://wxpay/bizpayurl?pr=uduaLGe","sign":"dpYZPOzdeTbSC1crvwm4EevBU
7X1xU5ee3rAYSFv5hgAPjQxkTaE0bhkBPiuOLMi0Giq8yCD+5fZ\nQHutkr8c93FUo7cecS+T+Ejtq
UjUnnahmw7ZKBmWA+RxYPg2jNsWT+s1zltBADe7/pjSW2h8VYuE\nT+FrGB0I24lY3Ga/rmo="}
```

2. Overseas Wechat is scanned directly, fixed payType = 71 issuerId = alipay, interface returns JSON string synchronously, such as

```
{"respCode":"T000","respMsg":"成功
","code_url":"https://qr.alipay.com/bax09734mfeogypsgn6e0058","sign":"aBuEgBgrcxxcW2/d
UysYEf3yxN6GgbAXECi295+HWTTXjXho79D0fsFIuErkNA9WbOUZm3jAI400\nCAVcUV27F9iMSN
XMi8RkYtsy3A0vGqpZfC+FAi/C7tPqSecBsHOJn7OxGbriaNSfu7vNuZ2CxlpO\nF7quVW2OJYnArM
ckc00="}
```

海外支付宝扫码直连，固定填写 payType=71 ,接口同步返回 json 字符串如

```
{"respCode":"T000","respMsg":"成功
","code_url":"https://qr.alipay.com/bax09734mfeogypsgn6e0058","sign":"aBuEgBgrcxxcW2/d
UysYEf3yxN6GgbAXECi295+HWTTXjXho79D0fsFIuErkNA9WbOUZm3jAI400\nCAVcUV27F9iMSN
XMi8RkYtsy3A0vGqpZfC+FAi/C7tPqSecBsHOJn7OxGbriaNSfu7vNuZ2CxlpO\nF7quVW2OJYnArM
ckc00="}
```

3. Wechat MiniProgram directly, fixed ext1=openid, payType = 80, issuerId = wechat, interface returns JSON string synchronously, such as

```
{"respCode":"T000","respMsg":"success","timeStamp":"1555580715","nonceStr":"MXhl6EmRq
1ZavLun","package":"prepay_id=wx181745150189642e0caf83670862708042","signType":"MD
5","paySign":"101D244A301126424BEC37742212D069","sign":"QCN5IJNjZqUSxUNa/eXPDaQm
e9WUfbXLIE75PbMR+PvMviYrVmJ93MC7MiMqF0oXd5A4bzOt1fwJ\r\nAJYo6bBRdUli8YpJ12w
W6UJoWi3bckCCrp+5djqo8gFf4IrOwMyfv4wi9d+87MLgfZeCdC/ay+pD\r\n3NF59BoNBbMfB4T+
9Ig="}
```

The merchant receives the return parameter and initiates the Wechat Payment，Please refer to：“https://pay.weixin.qq.com/wiki/doc/api/wxa/wxa_sl_api.php?chapter=7_7&index=5”

微信小程序直连，ext1=openid,payType = 80, issuerId = wechat,接口同步返回 json 字符串如下:

{"respCode":"T000","respMsg":"success","timeStamp":"1555580715","nonceStr":"MXhl6EmRq1ZavLun","package":"prepay_id=wx181745150189642e0caf83670862708042","signType":"MD5","paySign":"101D244A301126424BEC37742212D069","sign":"QCN5IJNjZqUSxUNa/eXPDaQme9WUfbXLIE75PbMR+PvMviYrVmJ93MC7MiMqF0oXd5A4bzOt1fwJ\r\nAJYo6bBRdUli8YpJ12wW6UJoWi3bckCCrp+5djqo8gFf4IrOwMyfv4wi9d+87MLgfZeCdC/ay+pD\r\n3NF59BoNBbMfB4T+9Ig="}

商户接收返回参数后可以向微信发起支付请求，请参考:https://pay.weixin.qq.com/wiki/doc/api/wxa/wxa_sl_api.php?chapter=7_7&index=5

4. WeChat official account directly, fixed ext1=openid, payType = 86, issuerId = wechat, interface returns JSON string synchronously, such as

{"respCode":"T000","respMsg":"success","timeStamp":"1555580715","nonceStr":"MXhl6EmRq1ZavLun","package":"prepay_id=wx181745150189642e0caf83670862708042","signType":"MD5","paySign":"101D244A301126424BEC37742212D069","sign":"QCN5IJNjZqUSxUNa/eXPDaQme9WUfbXLIE75PbMR+PvMviYrVmJ93MC7MiMqF0oXd5A4bzOt1fwJ\r\nAJYo6bBRdUli8YpJ12wW6UJoWi3bckCCrp+5djqo8gFf4IrOwMyfv4wi9d+87MLgfZeCdC/ay+pD\r\n3NF59BoNBbMfB4T+9Ig="}

The merchant receives the return parameter and initiates the Wechat Payment，Please refer to：“https://pay.weixin.qq.com/wiki/doc/api/wxa/wxa_sl_api.php?chapter=7_7&index=5”

微信公众号直连，ext1=openid,payType = 86, issuerId = wechat,接口同步返回 json 字符串如下:

{"respCode":"T000","respMsg":"success","timeStamp":"1555580715","nonceStr":"MXhl6EmRq1ZavLun","package":"prepay_id=wx181745150189642e0caf83670862708042","signType":"MD5","paySign":"101D244A301126424BEC37742212D069","sign":"QCN5IJNjZqUSxUNa/eXPDaQme9WUfbXLIE75PbMR+PvMviYrVmJ93MC7MiMqF0oXd5A4bzOt1fwJ\r\nAJYo6bBRdUli8YpJ12wW6UJoWi3bckCCrp+5djqo8gFf4IrOwMyfv4wi9d+87MLgfZeCdC/ay+pD\r\n3NF59BoNBbMfB4T+9Ig="}

商户接收返回参数后可以向微信发起支付请求，请参考:https://pay.weixin.qq.com/wiki/doc/api/wxa/wxa_sl_api.php?chapter=7_7&index=5

5. Wechat app directly, fixed ext1=openid, payType = 81, issuerId = wechat, interface returns JSON string synchronously, such as

{"respCode":"T000","respMsg":"success","timeStamp":"1555580715","nonceStr":"MXhl6EmRq1ZavLun","package":"prepay_id=wx181745150189642e0caf83670862708042","signType":"MD5","paySign":"101D244A301126424BEC37742212D069","sign":"QCN5IJNjZqUSxUNa/eXPDaQme9WUfbXLIE75PbMR+PvMviYrVmJ93MC7MiMqF0oXd5A4bzOt1fwJ\r\nAJYo6bBRdUli8YpJ12wW6UJoWi3bckCCrp+5djqo8gFf4IrOwMyfv4wi9d+87MLgfZeCdC/ay+pD\r\n3NF59BoNBbMfB4T+9Ig="}

The merchant receives the return parameter and initiates the Wechat Payment，Please refer to：“https://pay.weixin.qq.com/wiki/doc/api/wxa/wxa_sl_api.php?chapter=7_7&index=5”

微信 app 支付直连，ext1=openid,payType = 81, issuerId = wechat,接口同步返回 json 字符串如下:

{"respCode":"T000","respMsg":"success","timeStamp":"1555580715","nonceStr":"MXhl6EmRq1ZavLun","package":"prepay_id=wx181745150189642e0caf83670862708042","signType":"MD5","paySign":"101D244A301126424BEC37742212D069","sign":"QCN5IJNjZqUSxUNa/eXPDaQme9WUfbXLIE75PbMR+PvMviYrVmJ93MC7MiMqF0oXd5A4bzOt1fwJ\r\nAJYo6bBRdUli8YpJ12wW6UJoWi3bckCCrp+5djqo8gFf4IrOwMyfv4wi9d+87MLgfZeCdC/ay+pD\r\n3NF59BoNBbMfB4T+9Ig="}

商户接收返回参数后可以向微信发起支付请求，请参考:https://pay.weixin.qq.com/wiki/doc/api/wxa/wxa_sl_api.php?chapter=7_7&index=5

6. enets 扫码直连,固定填写 payType=69 issuerId = enets,接口同步返回字符串

{"respCode":"T000","respMsg":"success","txnReq":{"ss":"1","msg":{"netsMid":"UMID_867356001","tid":"","submissionMode":"B","txnAmount":"1000","merchantTxnRef":"CBO20190314210228","merchantTxnDtm":"20190314 16:36:01.827","paymentType":"SALE","currencyCode":"SGD","paymentMode":"QR","merchantTimeZone":"+8:00","b2sTxnEndURL":"http://119.23.136.80/ITSBoss/enetsPayNotifyAction/qrpageNotify.do","b2sTxnEndURLParam":"","s2sTxnEndURL":"http://119.23.136.80/ITSBoss/enetsPayNotifyAction/qrbackNotify.do","s2sTxnEndURLParam":"","clientType":"W","supMsg":"","netsMidIndicator":"U","ipAddress":"127.0.0.1","language":"en"}},"keyId":"524b1a17-3dfd-431b-b4cf-c20d86e07366","hmac":"DUaRXcSwRuHMRXZdegy7zsZ8E3gBA6n761eKHT1koDA="}

A.引入 enets 所需的 js 文件

```
<script src="https://www2.enets.sg/GW2/js/jquery-3.1.1.min.js"
type="text/javascript"></script>

<script src="https://www2.enets.sg/GW2/pluginpages/env.jsp"></script>

        <script                              src="https://www2.enets.sg/GW2/js/apps.js"
        type="text/javascript"></script>
```

B.调用此函数

sendPayLoad(txnReq, hmac, keyId);

C. 此 div 用于存放 enets 码

```
       <div style="width:200px; margin:0 auto; text-align:center;">

       <div id="anotherSection" style="height: 500px;width: 80%;padding-top:60px;padding-
right: 200px;">'

       <fieldset>

          <div id="ajaxResponse"></div>';

           </fieldset>

       </div>
```

7. Enets POS 扫码直连,固定填写 payType=90 issuerId = enets,接口同步返回字符串

{"respCode":"10000","respMsg":"SUCCESS","code_url":"iVBORw0KGgoAAAANSUhEUgAAAPoAAAD
6CAYAAACI7Fo9AAAMeElEQVR42u3dbXaiMBQAUPbjT9fiFlxHt+EWuhb+dj2Otk6LELRRkjzsfef04PjBI
OYm8EhCdxRCvHx0doEQoAshQBdCgC6EAF0IAboQAnQhBOhCCNCFAF0IAboQAnQhBOhCCNCFEKAL
IUAXQoAuBOhCCNB/sbKuq/p37/+vvjMX3v7Sny+9f2tv77PfL3r5Bh100EEHHXTQQQcddNBBf2ZHlIa0
9Odrw4he0T27/aX3R+mKd23lG3TQQQcddNBBBx100EEHveYPuXTBi5Z8K709pSvOpZOHS3/faMlF0E
EHHXTQQQcddNBBBx30V4Zeu2KonSxqvb9KJ5dqNwSlXwcddNBBBx100EEHHXTQQZeMi9MBJVoyrv
agiuiDTtbekIEOOuiggw466KCDDjrooNdM5pRO7rSG2Xr/rD052Xr/RPv9QAcddNBBBx100EEHHXTQSy
aD1t5hwuuv/brJIUH3Ouigg+510EEH3eugR4a+9mRU6eTS0snFV0/GlW5Y/lKADjrooIMOOuiggw466K
C3hvbqycToN7EsDbF2B5jWyUDQQQcddNBBBx100EEHHfSSUKIlk0r/cK07WESfeKJ2+WpdUdZMDoIO
Ouiggw466KCDDjrooD/7Rdc2cUS0mwxGq/haV5StGxbQQQcddNBBBx100EEHHfTIybhoHWKeLZilo/X
EHLXLw9K/R/RBRKCDDjrooIMOOuiggw466MdA0XpQyNLbU7riKV2wot0ksXTDUrs8/NnJIUEHHXTQQ
QcddNBBBx30Jsm40smI1gWjdDKn9cQKpQt+7f3zaslh0EEHHXTQQQcddNBBBx30khve+odY++SEa0u
OLl3R1K4ISm8/6KCDDjrooIMOOuiggw56Tehrv7F964Kzths2lK7ooneY+jM3cAAddNBBBx100EEHHXT
QQ0CPDjHaBPytK5rWybzSsEpXBNGSyaCDDjrooIMOOuiggw466C2TD2sbNLL0+qJ36Ii2fdHeXzNABx1
00EEHHXTQQQcd9L8HvXWyZG0TBdROrtXu8LN0sipawxFt+0EHHXTQQQcddNBBBx100Fsm66LddLD1
9tYumK1/z9rQok18AjrooIMOOuiggw466KCDvmSyqPYPWfuHiDZoJdrkl60H6awt2Qg66KCDDjrooIM
OOuiggx4p+Va7YK59oopoFXW0ZG3p/69l8g30ii2GiFWeQAcddNBBBx100EEvtuNrJ+/Afi3opf+/1jkb0
EEHHXTQQQcddNBBBx30yMmsKvDfDqe//WV5+dvtH9/owfr609/+vT/u3/rj4bI8//swWg5fv/W+friN4
21ey79P32Op8rbmZBroNaGfQPfn5emv77aX5c/j3Og3++91HC/r2Hf7Y7c5LPK3Pf0db2zveTn8PtPH28
v75x6n1llm/cfzvvroQQe9FvTtpSBuLzi33wXz83FObLaT9e1PBbrb9V9YJ8sz4H60nH/ff+ipbV7b4/+VxG
crDzrodaB3g0I4gpRzGL/ZD1rzQYt+BfaQgH1IwJ6+b7s7TLZ1+ribVlZXj7sBvNTjmuu/fO79AHpL6KWTdc
2Tb7t9soWcFOwb55XjFv26Beu+WvSbh+T9Lw/d+8Gh+/BQOVE5zX6n1OvdL55/5vPp9UyeO7a/Ceea
krugZ/wQ/Sf0OeCj57IO3X8K/OEE/Zxc2771M8vDndd/lp/bez4KuSzHf1fPT/B1P0coM+tJf36Kdm4b+n
v/Hp6jj44CzufroINeoUVPt0xXj3+ZjLuqIN4PTQpCv/vZlu/tefv9tgz3ySOfv11wEhXpad2gg14Mej8uzCeY
U7CX1mezzTp0XxRHbkwSjV1eviFxOjDOkj+/bd1VLgT0INBr75ji46evMHRXLXA/dwh/o4X+uby2/b4M1
gx64jSif6BFv9o/S32Xt/3o8P13625d/iJ1oAE9G/rM4enH8acQDl7vb7Vsm0SyKUCL/g0qq0VPXR7rvk4J
3hdo2c8V5m7QoWYF5Q/0FUOfnJcPC925d9uoZR9mie+3ot1nMu582ex8aWz762X6tTzoiaORXUaL/n
aYJvKu9lWXeH1uefncef88UVGADvrTLfowKTR/bfw6cXf/8tr25/La7vmecXmH7onz4Ld97g8wPSq4edk
utZy7/n45OgC9PfTWA/uLT6ifuoyUOgefK8CjSuH6HD3VYWbU+23y+LIcdpzZ9Y9BT1Viu8x1nPvXz+Uq
kp1kph1sbl++7K6OjtY28cSfHb22RuiTFmfufDF5ODrqTLOZtnxfLfoA9ezjUas/7iWXCX14ee17ex8ZrHPO
R8z0fEt3gpnrIHPjc5st6KBXbtHnoA9at8m56t0WfZlBLY8dug/zD0+MyjuDT1ylmC63M893NyqMDnTQ
S0JPHE7eygB/VwzTw9Iv6InunafPHD6On8NP+/fT8uM0vPw06vnZ97zfnn+Iz/rPsG0a3QFYYFhZb
Pbpa/S7A+hrgd46Gfbv5k5Vx1181z82nNm9Il+91nQzx2HMnsG5p1WjFr1B04rWk9OWruh
Av1p6Jkw7w4auZHFrxWJ3md9xqF7Oh+x0HdJJPnu9joEHfTnztG7/C6rH8fppba5a8fNesY9mXXfpDrLL
NUz7pA4rdCig15ofX1OMi55aDtoLScTV/xv0fdtoKd6/eVgmhmr/7l/nunzfr5KkZo8A/R20J+FHe3G9/cS
VtmjzSaYpr3olsy6b3eHB6AOLx1mYPqYv4Y+7qee83w/N5nFL3rLRSt/LQe5gJ49em2ENPdQe/YacTeCn
po9JmN5efz7ZNx0yGz2offfdcfr5PeTSk090qyx/oK+0RX80S54+Vx/MMDOeI27YIWY4Z9zc8//njMtq0R
MdWXaPfLdutlNMslvw+AjpTq+6nH0OOugPQl8oS/7Rz07aeOj2CeS/a72fadGPoyGz/YPnwf9b9uvD8
W3GVFW3+snnVT6gV4JeOllW+oeez/4Oem89mmj6Ts5dr+9X88V1c/PH9YPX8kaw9e+H6Xnzs0m0z
WD8+N3eb93ovYlRb3e2p2Wyq0TFAnor6IWjP4qaV31AB90tmUAHHXQBOuiL7djWyTfwXxt67ZuCtm
z4QAcddNBBBx100EEXoP896MUnfij8uuTcumHXhl69wxbooIMOOuiggw466GvJAdTOeSwNrfb3Ab1S
Mq427NIFozacaMmmaBVZ6+RY6YkqQAcddNBBBx100EEHHXTQS2ZBaw8qiJaMiZYcLF2RrW17a5d/0
EEHHXTQQQcddNBBBx30SMm4pTtoRCu4tZONr5aMjJbsrL3/QQcddNBBBx100EEHHXTQWybDaneYa
D0xxqt10Gg9qKN2Rb2m0Y6ggw466KCDDjrooIMOOuitk2W1kyHROtS8OpRoyb/WDQ/ooIMOOuigg
w466KCDDvozBaV2wYs28UDp5Fm0irN2xbf2igZ00EEHHXTQQQcddNBBB73oRfvGEzNEh772Dhxrvylh
tPIIOuiggw466KCDDjrooIMeCXrrglo62Rht+0snB6NVbGub/BJ00EEHHXTQQQcddNBBBz0SnLVNNlh6e
1rfdHLtkzNGTz6DDjrooIMOOuiggw466KC3jFfusBAxmdS64l264qz9/tY3BAEddNBBBx100EEHHXTQQ
Y8EtXVyJvr21r7BROny0rriiTZxBeiggw466KCDDjrooIMOes0fsnZyp3XF1roiLV3Q194hqHWHK9BBBx10

0EEHHXTQQQcd9JoVQbTkS+1kTu1kUesbJpSGHW1//tlBLaCDDjrooIMOOuiggw76KqC3rjhab0+0iRtK
F8zWya7a3692chp00EEHHXTQQQcddNBBB71lMiNaRRB90Entgli7A0ntDkitvy/ooIMOOuiggw466KC
DDnpkSK0nV6xd8Gsny0o3BK0rgugVDeiggw466KCDDjrooIMOeskdGX3QQ7Qb27dOdkYbFNL6/19Ts
hh00EEHHXTQQQcddNBBB7019GiTH0ZPnr3aRBRrbyhABx100EEHHXTQQQcddNBfORm3dDItenIvWj
KrdjIq2kQZLSdGAR100EEHHXTQQQcddNBBX3xjg09OGH3QSvT9Vfv9rZOxNf2ADjrooIMOOuiggw466
KBHS+7UTo6VLhivnoyLflPG0usDHXTQQQcddNBBBx100EEXQrxegC4E6EII0IUQoAshQBdCgC6EAF0IA
boQAnQhQBdCgC6EAF0IAboQAnQhBOhCCNCFEKALAboQ4lXjH44FL8KsaR4TAAAAAElFTkSuQmCC",
"pay_type":"90","signMsg":"592C438E8996CC46BE0F97BC273B43A3"}

获取接口同步返回中的 code_url

```
String code_url=jsonObject.getString("code_url") ;
```

将 code_url 传到页面，并在页面获取

```
var url ='<%=code_url%>';
```

显示二维码

```
<img id="enetsurl" src=""/>
$("#enetsurl").attr("src","data:image/png;base64,"+ url);
```

8. Overseas NextPos is scanned directly, fixed payType = 82 issuerId = NextPos, interface returns JSON string synchronously, such as

{"respCode":"T000","respMsg":"success","code_url":"00020101021230610016A0000006770101
12011501055510923800102121906010473850302DY530376454040.945802TH6211070787587
666304519E","sign":"i/eM6INrLlMbd0kyYVWYDTPKTr35qZg9d2SXnJwDK+L6tQsuKDs866KEoNah
tgtD6N/6QVRt1EFF\n5UcUOJvlYqcoJShAbqJjZCiJIs4GrYiVKr6Dj25giLGK7kPJd+LebJLEPo2pJrEvAx0
6HuD0AFAm\n7e45yUDj+gaRGNft7BQ="}

海外 NextPos 扫码直连，固定填写 payType=82 issuerId=NextPos,接口同步返回 json 字符串如：

{"respCode":"T000","respMsg":"success","code_url":"00020101021230610016A0000006770101
12011501055510923800102121906010473850302DY530376454040.945802TH6211070787587
666304519E","sign":"i/eM6INrLlMbd0kyYVWYDTPKTr35qZg9d2SXnJwDK+L6tQsuKDs866KEoNah
tgtD6N/6QVRt1EFF\n5UcUOJvlYqcoJShAbqJjZCiJIs4GrYiVKr6Dj25giLGK7kPJd+LebJLEPo2pJrEvAx0
6HuD0AFAm\n7e45yUDj+gaRGNft7BQ="}

9. 支付宝小程序直连 ,payType = 135 issuerId = alipay,接口同步返回 url，在小程序上跳转到此 url 可唤起支付宝钱包来支付

Alipay mini program directly,fixed payType = 135 issuerId = alipay, interface returns JSON string synchronously

10. 支付宝 H5 直连 ,payType = 136 issuerId = alipay,接口同步返回 url，根据 url 可唤起支付宝钱包来支付

Alipay H5 directly,fixed payType = 136 issuerId = alipay, interface returns JSON string synchronously

11. Alipay+ H5 直接，payType = 206 issuerId = Alipay+H5,接口同步返回 url，根据 url 可唤起支付宝钱包来支付

Alipay+ H5 directly,fixed payType = 206 issuerId = Alipay+H5, interface returns JSON string synchronously

5. 附录 Attachment

5.1 支付机构代码 Institution code

适用支付方式 **Payment**	**64- EC-WP-MPM-USD 80-EC-WP-MNP-SGD 86-EC-WP-PAC-SGD 107-EC-WP-MPM-SGD**
机构名称 Name	代码 Code
微信扫码 Wechat QR code payment	wechat

适用支付方式 **Payment**	**50-EC-AP-WEB-SGD 136-EC-AP-H5-SGD 206-EC_A+_H5_SGD**
机构名称 Name	代码 Code
支付宝扫码 ALIPAY QR code payment	alipay

适用支付方式 **Payment**	**171-EC-A+TNG-MPM-SGD 172-EC-A+DN-WEB-SGD 173-EC-A+GC-MPM-SGD 174-EC-A+HK-MPM-SGD 175-EC-A+TM-MPM-SGD 176-EC-A+KKP-MPM-SGD**
机构名称 Name	代码 Code
国际支付宝 ALIPAY+	alipay

适用支付方式 **Payment**	**Online banking: 63-EC-ENETS-MPM-SGD 73-EC-SINO-WEB-SGD 74-EC-MGP-WEB-THB 75-EC-MGP-MPM-THB 78-EC-SINO-WEB-CNY 79-EC-SINO-WEB-USD 129-EC-UM-CC-WEB-MYR 130-EC-UM-DD-WEB-MYR 139-EC-PNOW-MPM-SGD 152-EC-GPAY-WEB-SGD 159-EC-GPAY-WEB-SGD**

	161-EC-GPAY-WEB-SGD
机构名称 Name	代码 Code
东亚银行	HKBEA
Bank Negara Indonesia	BNI
大通曼哈顿银行	CMBANK
第一洲际银行	FIB
汉华实业银行	MHC
梅隆国民银行	MNC
美洲银行国民信托储蓄会	BAC
摩根保证信托银行	MGTCNY
纽约化学银行	CNYC
纽约银行家信托公司	BTNYC
太平洋安全银行	SPC
Affin Bank	FPXD_ABB0233
Alliance Bank	FPXD_ABMB0212
Ambank	FPXD_AMBB0209
Bank Islam	FPXD_BIMB0340
Bank Muamalat	FPXD_BMMB0341
BSN	FPXD_BSN0601
CIMB	FPXD_BCBB0235
Hong Leong Bank	FPXD_HLB0224
Kuwait Finance House	FPXD_KFH0346
Maybank	FPXD_MB2U0227
OCBC Bank	FPXD_OCBC0229
Public Bank	FPXD_PBB0233
RHB	FPXD_RHB0218
Standard Charted Bank	FPXD_SCB0216

Permata Bank	permata
Standard Bank Isle of Man Limited	SBIOM
Bank of Ayudaya	BAY
Bangkok Bank	BBL
Krung Thai Bank	Krung_Thai_Bank
Siam Commercial Bank	SCB
Thai Military Bank	TMB
Kleinwort Benson Bank	KBB
City National Bank	CNB
Vietcom Bank	VCB
CIMB-Thai	CIMBTH
Bank Central Asia	BCA
Bank ¨¢ Chau AC	ACB
Bank Đầu tư và Phát triển Việt Nam BIDV	BIDV
Bank Đông Á	DongABank
Bank Hàng Hải Việt Nam Maritime Bank	MaritimeBank
Bank Quân Đội MB	MB
Bank Sài Gòn Thương Tín Sacombank	Sacombank
Bank Kỹ thương Việt Nam	Techcombank
Bank Ngoại Thương Việt Nam	Vietcombank
Bank Công thương Việt Nam Vietinbank	Vietinbank
RHB Banking Group	RHB
AmBank Group	AMB
Bank Islam Malaysia	BIMB
Karsikorn Bank (K-Bank)	KKR

Bank of Ayudhya (Krungsri)	BOA
Government Savings Bank	GSB
Kiatnakin Bank	KNK
Techcombank	TCB
Sacombank	SACOM
Eximbank Vietnam	EXIM
Bank Rakyat Indonesia	BRI
Mandiri Bank	MDR
CIMB Niaga	CIMBN
Banco de Oro	BDO
MetroBank	MTB
eNets	eNets
NextPos	NextPos
HONGKONG AND SHANGHAI BANKING CORPORATION LIMITED	004
GrabPayBank	GRAPB
Thai_Qr_Bank	TQB
finexusPaymasterBank	FINXPMB_CC
maybank2u	maybank2u
FPX_HSBC	FPX_HSBC
MB2U	MB2U
PBB	PBB
AMOnline	AMOnline
HLBConnect	HLBConnect
RHBNow	RHBNow
FPX_OCBC	FPX_OCBC
FPX_SCB	FPX_SCB
FPX_ABB	FPX_ABB

bankrakyat	bankrakyat
bankmuamalat	bankmuamalat
FPX_KFH	FPX_KFH
FPX_BSN	FPX_BSN
FPX_ABMB	FPX_ABMB
FPX_UOB	FPX_UOB
CIMBCLICKS	CIMBCLICKS
Citi	CITISGSGXXX
DBS Bank	DBSSSGSGXXX
OCBC	OCBCSGSGXXX
Maybank Singapore	MBBESGSGXXX
Standard Chartered Bank	SCBLSG22XXX
United Overseas Bank	UOVBSGSGXXX
CIMB Bank	CIBBSGSGXXX
HSBC	HSBCSGS2XXX
RHB Bank	RHBBSGSGXXX
ICBC	ICBKSGSGXXX
RAZER Bank	RAZER
PayNow	PayNow
Atome Bank	ATOME
dokuBank	dokuBank
UpiBank	UpiBank
ShopeePay_Bank	SHOPEE
PayNowDBS	PayNowDBS
MEGAPAY_UOB	UOB

5.2 业务类型代码 Code of business type

业务类型 **Business type**	
业务类型名称 Name	代码 Code
人民币业务 CNY business	1
跨境业务 Cross-border business	2

附测试环境及生产环境商服系统登录信息：
TEST ENVITONMENT AND LIVE ENVIRONMENT LOGIN INFO：

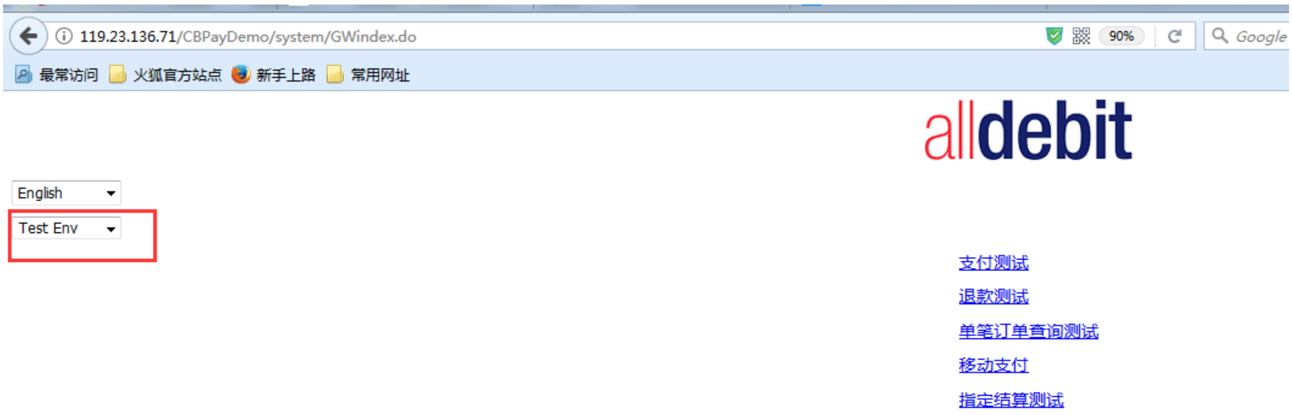
测试环境：
Test Environment:

MID：20170921500044
User ID：test
password：test123
Transaction PIN：test123

Test Merchant Service platform: (where you can login to get keys)
http://test-mer.alldebitpay.com/system/login.do

If need to useDEMO in the test environment, please use the link as below.

http://test-demo.alldebitpay.com/system/GWindex.do



119.23.136.71/CBPayDemo/system/GWindex.do

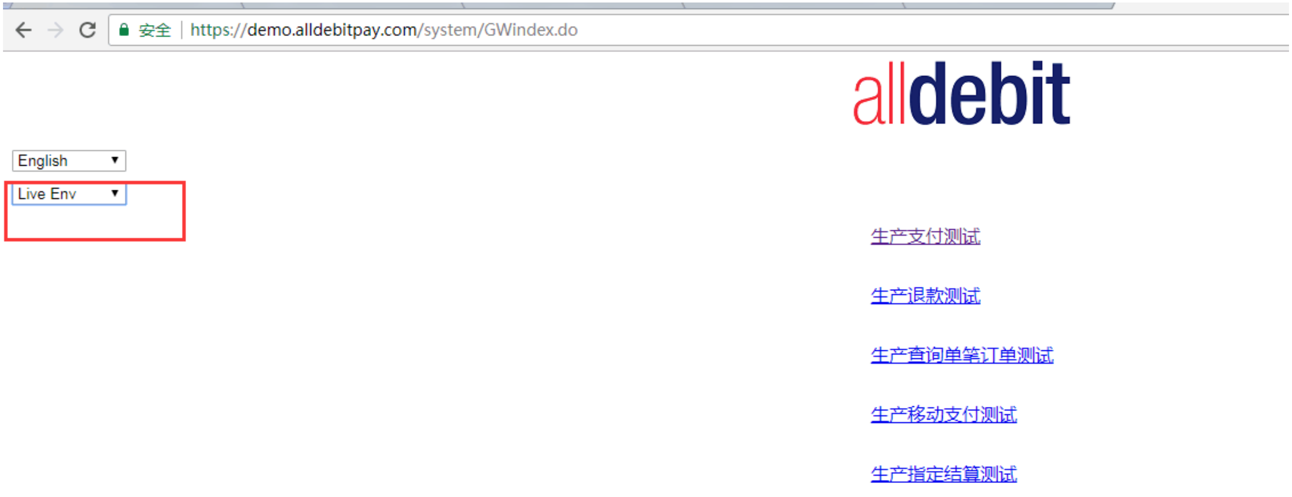
English
Test Env

支付测试
退款测试
单笔订单查询测试
移动支付
指定结算测试

生产环境：
Live Environment:
请联系我们开出商户号。
Please contact us to issue a live MID to you.

备注：对应商户支付需要的商户秘钥和平台公钥请登录商服平台获取。
Remark: merchant can get the merchant private key and platform public key after login to merchant service

platform.
若使用到Demo，对应的地址如下：
If need to useDEMO in the live environment, please use the link as below
https://demo.alldebitpay.com/system/GWindex.do



安全 | https://demo.alldebitpay.com/system/GWindex.do

English

Live Env

生产支付测试

生产退款测试

生产查询单笔订单测试

生产移动支付测试

生产指定结算测试

注：要注意切换到对应测试环境或生产环境再进行操作。
When in test environment, please switch to test environment;
When in live environment, please switch to live environment.

5.3 开发者参考 Reference for developer

注：使用 RSA 公钥和私钥签名前必须去掉密钥字符串里的空格【针对以下所有情况的开发】

Remark: Remove allthe blanks in the character strings of keys when using RSA Public key and private key to sign (Aim at the following developing languages.)

5.3.1JAVA 开发者参考

For JAVA : None

5.3.2PHP 开发者参考

For PHP

1) 签名注意事项：

Attention of signature:

使用 PHP 开发签名部分的时候请使用 openssl_sign 来处理接口中的 RSA 签名方式：

Use openssl sign to deal with the RSA signature method when developing signature part when using PHP

例如：

Example:

string = ‘orderId123456’;

openssl_get_privatekey($priv);

openssl_sign ($string, $signature, $priv, OPENSSL_ALGO_MD5);

请重点注意红色标注的内容。Please pay attention to the words in red.

参照网址：http://www.zhimengzhe.com/bianchengjiaocheng/Javabiancheng/126918.html

Refer to :http://www.zhimengzhe.com/bianchengjiaocheng/Javabiancheng/126918.html

2) 其他注意事项: 暂无

Others: None.

5.3.3.NET 开发者参考

For .NET：

1) 签名注意事项：

Attention of signature:

使用 C#开发签名部分的时候请使用 PKCS8 来处理接口中的 RSA 签名方式：

Use PKCS8 sign to deal with the RSA signature method when developing signature part when using C#

核心代码类如下：

Example:

RSAFromPkcs8.cs

详细 demo:

ad3_demo.zip

2) 其他注意事项: 暂无

Others: None.