
Brighte Documentation

Release 1.1

Brighte Capital

Sep 14, 2018

Contents

1	Checkout API	1
1.1	Development	1
1.1.1	Introduction	1
1.1.2	Implementation	1
2	Webhooks	6
2.1	Verifying request	7
3	Sandbox	8
3.1	Simulated behaviour	8
4	Restful API	9
4.1	Initial Setup for the Rest API	9
4.1.1	Session keys	9
4.1.2	Content Type	9
4.1.3	Date and time formats	10
4.1.4	Account Specifier	10
4.2	Sessions	10
4.2.1	Create Session	10
4.2.2	Delete Session	11
4.3	Accounts	11
4.3.1	Get Accounts	11
4.4	Applications	12
4.4.1	Get Applications	12
4.4.2	Get Application	13
4.4.3	Update Application	14
4.4.4	Get Application Document Types	15
4.4.5	Get Application Documents	16
4.4.6	Create Application Document	17
4.4.7	Request Application Payment	18
4.4.8	Send Application Secure Code	19
4.5	Quotes	20
4.5.1	Get Quotes	20
4.5.2	Get Quote	21
4.5.3	Create Quote	22
4.5.4	Resend Quote	23
4.5.5	Delete Quote	24

4.6	Users	25
4.6.1	Get Users	25
4.6.2	Get User	26
4.6.3	Create User	27
4.6.4	Update User	28
4.6.5	Delete User	29
4.6.6	Generate User Api Credentials	30
4.7	Content	31
4.7.1	Get Content	31
4.8	Product Categories	32
4.8.1	Get Product Categories	32
4.9	Calculate Repayment Amount	33
4.9.1	Calculate Repayment Amount	33
4.10	Callout Requests	34
4.10.1	Create Callout Request	34
5	API Data Types	36
5.1	Simple Data Types	36
5.2	Enums	37
5.3	Complex Data Types (aka POD objects)	37
5.3.1	Application	37
5.3.2	Applicant	38
5.3.3	ApplicationDocument	38
5.3.4	Quote	39
5.3.5	User	39
5.3.6	Account	40
5.3.7	Content	40
5.3.8	ProductCategory	40
5.3.9	RepaymentAmount	40
5.3.10	Callout	41
5.3.11	UserKey	42

Brighte Checkout Application Programming Interface (API) allows the integration of creating loan applications from your server. This gives your business a self-serve experience for your customers to purchase your goods and services with Brighte.

1.1 Development

For a quick introduction, [view an example on our demo environment](#) or use the demo API endpoint `https://demo.brightelabs.com.au/checkout?key=0d5dd2d57ee0fb9973bde47cef60a2ba`. You may also request a fully-functional *sandbox environment* from Brighte.

1.1.1 Introduction

Brighte allows vendors to create loan applications for their customers. Using the Checkout API, loan applications can be initialised from a vendor web application and completed by the customer on Brighte.

The Checkout solution allows the vendor to direct the customer to Brighte with an API request. The API request from the vendor contains the financing amount and product description. The customer completes a simple and fast loan application process. The customer is redirected back to the vendor web application with an API response containing confirmation of the application.

1.1.2 Implementation

This section details the process of integrating the Checkout API for your business.

Setup

You must first have a vendor account with the Checkout API enabled. Follow these steps:

- Login to the Vendor Portal at `https://portal.brighte.com.au/login`.

- From the Developer area, click the API keys section.
- Create a new public API key with restrictions for referrer and redirect URLs.
 - The referrer URL restriction specifies the URLs from which API requests are accepted.
 - The redirect URL is a page provided in the API request and to which the customer is redirected after an application. API requests providing invalid redirect URLs will not be accepted.
- You can also create a private key if you want to authenticate the origin of the request.
- Integrate the Checkout API into your web application.

API request

API requests are HTTP requests sent by the customer. They must meet the following conditions:

- The request is sent to `https://portal.brighte.com.au/checkout?key=YOUR_API_KEY`
- The request must be sent via POST method.
- The request is sent from a URL conforming to the restrictions set on the public API key.
- The redirect URL must conform to the restrictions set on the public API key. This URL can contain a query string component can be used to store parameters between sessions.
- The deposit amount and repayment term must be valid according to your vendor account.
- The product category must be one of the following:
 - Storage Battery(s)
 - Solar Panel Package
 - Solar Panel and Battery Combo
 - Solar Inverter(s)
 - Flooring
 - Guttering
 - Roofing
 - Air-Conditioning
 - Window Shutters
 - Hot Water Systems
 - Pool Heating Systems (Solar/Pump/Gas)
 - Smart Home Technology
 - Trailers and Campers
 - Carports/Patio/Pergolas
 - Doors/Garage Doors
 - Home Improvements
 - Windows/Blinds/Shutters/Glazing
 - Plumbing and Electrical
 - Heating/Cooling Solutions
 - Security System

Required paramters:

Parameter	Data type	Description
redirect	string	Redirect URL upon application completion
total_purchase_amount	double	Purchase total
deposit_amount	double	Deposit amount
repayment_term	integer	Repayment term in months
product_category	string	Product category
product_description	string	Product description
reference_number	string	Optional reference number (max 20 chars)
checksum	string	SHA-256 hash of the POST parameters in the order listed below

The checksum parameter must be a SHA-256 hash of a concatenated string in this order

- redirect
- total_purchase_amount
- deposit_amount
- repayment_term
- product_category
- product_description

JSON-formatted sample request:

```
{
  "redirect": "http://yourwebsite.com/checkout-success?yourtrackingid=S888",
  "total_purchase_amount": "7000.12",
  "deposit_amount": "750.24",
  "repayment_term": "24",
  "product_category": "Storage Battery(s)",
  "product_description": "Powerwall Home Battery",
  "reference_number": "12345J",
  "checksum": "5bf7eef1a95160f47134717996d6c5ae"
}
```

Sample HTML code that sends an API request:

```
<html>
<head><title>Some Solar Purchase</title></head>
<body>
  <h1>Some Solar</h1>
  <h2>Powerwall Battery: $7,000</h2>
  <p></p>
  <h3>Buy now with Brighte</h3>
  <h4>[ Include full details and description of item that makes up the price here ]
  </h4>
  <ul>
    <li>Pay with interest free payment plans.</li>
    <li>Fees apply.</li>
    <li>Repayments of $136.05 per fortnight.</li>
    <li>Total repayable $7,314.48.</li>
  </ul>
</body>
</html>
```

(continues on next page)

(continued from previous page)

```

</ul>
<form action="https://portal.brighte.com.au/checkout?key=YOUR_API_KEY" method=
↪ "POST">
  <input type="hidden" name="redirect" value="http://yourwebsite.com/checkout-
↪ success.html">
  <input type="hidden" name="total_purchase_amount" value="7000">
  <input type="hidden" name="deposit_amount" value="700">
  <input type="hidden" name="repayment_term" value="24">
  <input type="hidden" name="product_category" value="Storage Battery(s)">
  <input type="hidden" name="product_description" value="Powerwall Home Battery
↪ ">
  <input type="hidden" name="checksum" value="5bf7eef1a95160f47134717996d6c5ae">
  <button type="submit">Buy now with Brighte</button>
</form>
<p>*Terms, conditions and lending criteria apply. Repayments based on RRP of $7,
↪ 000 and 0% deposit. Minimum amount payable $7,314.48 over 24 months.<br>Fees and
↪ charges apply includes $75 Application Fee, $3.50 monthly Account Keeping Fee and
↪ $2.99 fortnightly Payment Processing Fee. Ask in-store for details or visit Brighte.
↪ com.au. Continuing credit provided by Brighte Ptd Ltd</p>
</body>
</html>

```

API response

If the API request has an invalid `redirect` value, a HTTP 400 error is returned. If the API request is invalid but has a valid `redirect` value, the customer is redirected to the redirect URL with details of the error provided through GET parameters `error_code` and `error_msg`. The error codes are as follows:

Error code	Meaning
10002	Invalid data given. See <code>error_msg</code> for more information.

After the customer submits an application successfully, the customer is redirected to the redirect URL with the following GET values:

Parameter	Value
<code>error_code</code>	10000
<code>transaction_id</code>	ID of the application. Unique numeric value.

Sample HTML and PHP code that receives an API response:

```

<html>
<head>
<title>Some Solar Purchase</title>
</head>
<body>
  <h1>Some Solar</h1>
  <?php switch ($_GET['error_code']) : ?>
  <?php case 10000 : ?>
  <h2>Payment application submitted successfully.</h2>
  <p>You should receive an email from Brighte to complete your application.</p>
  <p>Your transaction ID is: <?php echo htmlspecialchars($_GET['transaction_id']); ?
↪ ></p>
  <?php break; ?>

```

(continues on next page)

(continued from previous page)

```

<?php default: ?>
<h2>Payment application failed.</h2>
<p>Error: <?php echo htmlspecialchars($_GET['error_code']); ?> - <?php echo_
↳htmlspecialchars($_GET['error_msg']); ?></p>
</body>
</html>

```

Securing webhook

As the redirect address is public, you may verify the request came from Brighte using a private API key. If you have created a private API key, extra GET parameters `signature`, `timestamp` and `token` are sent. Follow these steps to verify the request:

- Login to the Vendor Portal at <https://portal.brighte.com.au/login>.
- From the Developer area, Create a new private API key.
- Verify every request by concatenating the `timestamp` and `token` values and encoding the resulting string with the HMAC algorithm. Use the private API key as the key and SHA256 digest mode. The resulting HEX digest should match the `signature` value.

Sample PHP code that verifies an API response:

```

if ($_GET['signature'] !== hash_hmac('sha256', $_GET['timestamp'] . $_GET['token'],_
↳utf8_encode(BRIGHTE_PRIVATE_KEY))) {
    exit;
}

```


CHAPTER 2

Webhooks

Brighte uses webhooks to provide event notification. Webhooks are HTTP callbacks that receive notification messages of consumer application status updates. Webhooks can be created using the Vendor Portal or using the REST API.

When a change to the consumer application status occurs, Brighte issues an HTTP POST notification message to your webhook's URL. Your server is expected to return a status code of either 200 (Success) or 406 (Not Acceptable) to indicate the message has been successfully received. For any other code, the message will be resent for 8 hours at the following intervals before stopping: 10 minutes, 10 minutes, 15 minutes, 30 minutes, 1 hour, 2 hours and 4 hours.

The webhook is called with the following parameters:

Parameter	Description
id	Consumer application identifier
reference_number	Consumer application reference number
status	Consumer application status
timestamp	Number of seconds passed since 1 January 1970
token	Randomly generated string of length 50
signature	String with hexadecimal digits generated by HMAC algorithm

Sample payload:

```
{
  "id": "BRIGHTEID",
  "reference_number": "BRIGHTEREF",
  "status": "APPROVED",
  "timestamp": "946684800",
  "token": "b08c02e08efec9e34611c5141e5f698bff949f95999e687ecc",
  "signature": "4743b5be4b4eddebb7fc43d6bff001a067e1c7e11c69a2d022fb465e4e6ae4e8"
}
```

2.1 Verifying request

Your server must verify that notification messages originated from Brighte using the `signature` parameter.

To verify the authenticity of the message:

- Concatenate the `timestamp` and `token` values.
- Encode the string with the HMAC method using your Private API Key (encoded in a UTF-8 format) and the SHA256 algorithm.
- Compare the calculated message digest with the signature.

Brighte provides a sandbox environment to assist you with integration. Request your sandbox by contacting [Brighte Support](#).

3.1 Simulated behaviour

Status changes are simulated for all applications submitted to the sandbox. All submitted applications will have their statuses changed according to a fixed pattern.

If the last digit of the financed amount (purchase total minus deposit amount) is:

- 1, the status is changed to C . APPROV immediately, then to APPROVED after several minutes.
- 2, the status is changed to C . APPROV immediately, then to DECLINED after several minutes.
- 3, the status is changed to REFER immediately, then to APPROVED after several minutes.
- 4, the status is changed to REFER immediately, then to DECLINED after several minutes.
- 5, the status is changed to DECLINED immediately.

Example	Simulated behaviour
\$1001.00	C . APPROV immediately, then to APPROVED
\$1005.99	DECLINED immediately

4.1 Initial Setup for the Rest API

Brighte exposes some of the functionality required to create and manage applications, quotes and users to authorised users, via a Rest API.

4.1.1 Session keys

You need to acquire access keys in order to interact with the API. Please contact our [support team](#) that will happily assist you in procuring them.

After you have acquired your access keys, you will need to create a session. You will need to use the [create session](#) endpoint.

Every request will need to include the *X-Session-Key* and *X-Session-Token* headers.

Key	Value
X-Session-Key	string
X-Session-Token	string

4.1.2 Content Type

All endpoints should be able to accept json so you will need to add the following headers:

Key	Value
Accept	application/json
Content-Type	application/json

4.1.3 Date and time formats

The rest API uses the **UTC timezone** for both input and output parameter. The timestamps returned from this API use **ISO8601** format (e.g. `2016-08-08T06:54:36+00:00`) which includes the timezone as well.

4.1.4 Account Specifier

Most of the entities within the Brighte platform are associated with accounts (branches). Whenever you are trying to interact with these entities, you will need to specify an **account id** within the request. This documentation will tell you where to specify it and how.

4.2 Sessions

4.2.1 Create Session

First call needs to be to create a session by authorizing your user using the api key and the api secret.

Refer to the top of this guide to find out how to procure them.

Endpoint URL:

Method Type	Url
POST	portal.brighte.com.au/rest/v1/vendor/session

Parameters:

Name	Value	Description
apiId	string	This is the API KEY
apiSecret	string	This is the API SECRET

Sample request:

HTTP

```
POST /rest/vendor/v1/session HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
Cache-Control: no-cache

{
  "apiId": "xxx",
  "apiSecret": "xxx"
}
```

Successful response:

```
{
  "status": 2000,
  "sessionKey": "19469f463f1024d4f44c2441bd7d9f13a913b9d6",
  "sessionToken": "109e67ea15892d29ae5ecc241c3f2c399e293004"
}
```

Error codes:

Code	Status	Message	Reason
401	4010	Invalid credentials.	Wrong Key / Secret combination.
401	4010	Unauthorized.	User does not have access to the API.

4.2.2 Delete Session

Delete your session / log out the current user.

Endpoint URL:

Method Type	Url
DELETE	portal.brighte.com.au/rest/v1/vendor/session

Parameters:

No parameters

Sample request:

HTTP

```
DELETE /rest/vendor/v1/session HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful response:

```
{
  "status": 2000
}
```

Error codes:

Code	Status	Message	Reason
401	4010	Unauthorized.	User does not have access to the API.
400	4402	Invalid session.	Session not valid for deletion.
400	4010	Failed to delete session.	Session not found.

4.3 Accounts

4.3.1 Get Accounts

Get list of all the accounts associated with this user.

Endpoint URL:

Method Type	Url
GET	portal.brighte.com.au/rest/v1/vendor/accounts

Parameters:

No Parameters

Sample request:

HTTP

```
GET /rest/vendor/v1/account HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful response:

```
{
  "status": 2000,
  "accounts": [
    Account POD
  ],
}
```

Resources:

- [Account POD](#)

Error codes:

Code	Status	Message	Reason
404	4041	No account found.	No account set for current user, contact support.

4.4 Applications

4.4.1 Get Applications

Return all the applications.

The results are ordered by creation date. There is a maximum limit of 500 records per call and the next result set can be retrieved by utilizing the *pageNumber* parameter and the *hasMoreRecords* return field.

Endpoint URL:

Method Type	Url
GET	portal.brighte.com.au/rest/vendor/v1/applications

Parameters:

Name	Value	Description
accountId	string	Id of the account to list applications for.
startDate	date	(optional, default beginning) Start Date
endDate	date	(optional, default today) End Date
pageNumber	integer	(optional, default 1) Page Number

Sample request:

HTTP

```
GET /rest/vendor/v1/applications HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful response:

```
{
  "status": 2000,
  "applications": [
    Application POD
  ],
  "hasMoreRecords": boolean
}
```

Resources:

- [Application POD](#)

Error codes:

Code	Status	Message	Reason
401	4010	Not authorized.	You are not authorized to perform this action with given parameters.
400	4402	Invalid date input.	Wrong <i>startDate</i> or <i>endDate</i> input.
400	4402	Invalid application ID.	Please specify an application id.
404	4041	Invalid application ID.	Application not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid application ID.	You are not authorized to access this resource.

4.4.2 Get Application

Return information about a specific application.

Endpoint URL:

Method Type	Url
GET	portal.brighte.com.au/rest/vendor/v1/applications/{application_id}

Parameters:

Name	Value	Description
application_id	string	Id of the application

Sample request:

HTTP

```
GET /rest/vendor/v1/applications/{application_id} HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful response:

```
{
  "status": 2000,
  "application": "Application POD"
}
```

Resources:

- [Application POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Invalid application ID.	Please specify an application id.
404	4041	Invalid application ID.	Application not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid application ID.	You are not authorized to access this resource.

4.4.3 Update Application

Edit an existing application. It is only possible to update the *reference_number*.

Endpoint URL:

Method Type	Url
PUT	portal.brighte.com.au/rest/vendor/v1/application/{application_id}

Parameters:

Name	Value	Description
reference_number	string	New reference number of the application.

Sample request:

HTTP

```
PUT /rest/vendor/v1/application/{application_id} HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache

{
  "reference_number": "NEW_REF_NUM"
}
```

You should receive the following response.

Successful Sample Response:

```
{
  "status": 2000,
  "application": "Application POD"
}
```

Resources:

- Application POD

Error codes:

Code	Status	Message	Reason
400	4402	Invalid application ID.	Please specify an application id.
404	4041	Invalid application ID.	Application not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid application ID.	You are not authorized to access this resource.

4.4.4 Get Application Document Types

Retrieve all the possible document types for current user.

Endpoint URL:

Method Type	Url
GET	portal.brighte.com.au/rest/vendor/v1/application/document-types

Parameters:

No parameters

Sample request:

HTTP

```
GET /rest/vendor/v1/application/document-types HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful response:

```
{
  "status": 2000,
  "applicationDocumentTypes": [
    "invoice",
    "CCA"
  ]
}
```

Resources:

- `applicationDocumentType` ENUM

Error codes:

Code	Status	Message	Reason
500	4000	Internal Error.	Internal error, contact support.

4.4.5 Get Application Documents

Retrieve all the documents (invoices, ccas, stcs...) associated with a certain application.

Endpoint URL:

Method Type	Url
GET	portal.brighte.com.au/rest/v1/vendor/application/{application_id}/document

Parameters:

Name	Value	Description
application_id	string	Id of the application

Sample request:

HTTP

```
GET /rest/vendor/v1/application/{application_id}/document HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful response:

```
{
  "status": 2000,
  "applicationDocuments": [
    "ApplicationDocument POD"
  ]
}
```

Resources:

- [Application Document POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Invalid application ID.	Please specify an application id.
404	4041	Invalid application ID.	Application not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid application ID.	You are not authorized to access this resource.

4.4.6 Create Application Document

Upload documents (invoices, STCs, CCAs ...) to the given application.

Endpoint URL:

Method Type	Url
POST	portal.brighte.com.au/rest/v1/vendor/application/{application_id}/document

Parameters:

Name	Value	Description
application_id	string	Id of the application
invoice_type	string	Type of the document, one of applicationDocumentType ENUM
filename	string	Name of the file, allowed types are 'jpg', 'jpeg', 'png', 'pdf'
filecontent	string	BASE64 encoded content of the document

Sample request:

HTTP

```
POST /rest/vendor/v1/application/{application_id}/document HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache

{
  "invoice_type": "invoice",
  "filename": "invoice.jpg",
  "filecontent": "xxx"
}
```

Successfull response:

```
{
  "status": 2000,
  "applicationDocument": ApplicationDocument POD,
}
```

Resources:

- [applicationDocumentType ENUM](#)
- [Application Document POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Invalid application ID.	Please specify an application id.
404	4041	Invalid application ID.	Application not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid application ID.	You are not authorized to access this resource.
400	4402	Invalid application document type.	The document type has to be one of applicationDocumentType ENUM .
400	4010	No document type specified.	You have to specify the application document type via invoice_type parameter.
400	4010	Invalid filename.	You have to specify the document filename.
400	4010	Invalid filetype.	Allowed types are 'jpg', 'jpeg', 'png', 'pdf'.
400	4402	File content has to be base64 encoded.	Encode the content with BASE64.
500	4000	Internal Error.	Failure, contact support.

4.4.7 Request Application Payment

After the successful delivery and installment of the product at the physical location of the buyer by the installer, this method should be invoked to indicate that the buyer is ready to be started on the repayments.

Before he is actually started on the repayments however, the buyer is prompted to confirm the successful delivery and installment as well.

Endpoint URL:

Method Type	Url
POST	portal.brighte.com.au/rest/vendor/v1/application/{application_id}/request-payment

Parameters:

Name	Value	Description
application_id	string	Id of the application

Sample request:

HTTP

```
POST /rest/vendor/v1/application/{application_id}/request-payment HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful response:

Successful Sample Response:

```
{
  "status": 2000,
  "application": "Application POD"
}
```

Resources:

- [Application POD](#)

Error codes:

Code	Status	Message	Reason
401	4403	Unauthorized.	You are not authorized to access this resource.
404	4041	Not found.	Invalid application id.
500	5000	Error occurred.	There was an error with this request, contact support.
400	4402	Invalid application ID.	Please specify an application id.
404	4041	Invalid application ID.	Application not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid application ID.	You are not authorized to access this resource.

4.4.8 Send Application Secure Code

Send application secure code to the applicant's mobile phone. This API call will return a random token that will need to be submitted along with the secure code.

Endpoint URL:

Method Type	Url
POST	portal.brighte.com.au/rest/vendor/v1/application/secure-code

Parameters:

Name	Value	Description
mobile	string	The mobile phone number as a plain string without country code.

Sample request:

HTTP

```
POST /rest/vendor/v1/application/secure-code HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache

{
  "mobile" : "0416123456"
}
```

Successful response:

```
{
  "status": 2000,
  "token": "072b17a51edf4ca38f836293f3018f7117fa2c88"
}
```

Error codes:

Code	Status	Message	Reason
400	4403	Invalid Input.	You are not authorized to access this resource.
404	4040	Failed to generate auth code.	Failed to generate code, contact support.
500	4000	Failed to process auth code.	Failed to process code, contact support.
500	5000	Error occurred.	There was an error with this request, contact support.

4.5 Quotes

4.5.1 Get Quotes

Return all the quotes for a specific account.

The results are ordered by creation date. There is a maximum limit of 500 records per call and the next result set can be retrieved by utilizing the *pageNumber* parameter and the *hasMoreRecords* return field.

Endpoint URL:

Method Type	Url
GET	portal.brighte.com.au/rest/vendor/quotes

Parameters:

Name	Value	Description
accountId	string	Id of the account to get the quotes for.
startDate	date	(optional, default beginning) Start Date
endDate	date	(optional, default today) End Date
pageNumber	integer	(optional, default 1) Page Number

Sample request:

HTTP

```
GET /rest/vendor/v1/quotes?startDate=2018-04-01&endDate=2018-05-04&accountId=E897.675
↪ HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful response:

```
{
  "status": 2000,
  "quotes": [
    Quote POD
  ],
  "hasMoreRecords": boolean
}
```

Resources:

- [Quote POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Invalid date.	Wrong <i>startDate</i> or <i>endDate</i> input.
401	4010	Not authorized.	You are not authorized to perform this action with given parameters.
400	4402	Invalid account.	Wrong account id provided. See Get Accounts
500	5000	Internal error.	Internal error occurred, contact support.

4.5.2 Get Quote

Return information about a specific quote.

Endpoint URL:

Method Type	Url
GET	api.brighte/rest/vendor/quotes/{id}

Parameters:

Name	Value	Description
id	string	Id of the quote.

Sample request:

HTTP


```
GET /rest/vendor/v1/quotes/1 HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful response:

```
{
  "status": 2000,
  "quote": Quote POD
}
```

Resources:

- [Quote POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Invalid quote ID.	Please specify a quote id.
404	4041	Invalid quote ID.	Quote not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid quote ID.	You are not authorized to access this resource.

4.5.3 Create Quote

Create a quote.

You need to specify an account id.

Endpoint URL:

Method Type	Url
POST	portal.brighte.com.au/rest/vendor/v1/quote

Parameters:

Name	Value	Description
account_id	string	Account id of the branch that this quote is for.
first_name	string	First name of the quote receiver.
last_name	string	Last name of the quote receiver.
phone	string	Phone of the quote receiver.
email	string	Email of the quote receiver.
total_purchase_amount	currency	The total amount of the quote.
deposit_amount	currency	The deposit of the quote.
repayment_term	integer	Repayment term in months.
product_category	string	Category of the product in this application.
product_description	string	Description of the financed items.

Sample request:

HTTP

```
POST /rest/vendor/v1/quote HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache

{
  "first_name" : "Test First",
  "last_name" : "Lastnametest",
  "phone" : "0434632777",
  "email" : "generic.test.case420@gmail.com",
  "total_purchase_amount" : 15000,
  "deposit_amount" : 5000,
  "repayment_term" : 48,
  "product_category" : "Storage Battery(s)",
  "product_description" : "This product stores solar energy",
  "account_id" : "E435"
}
```

Successfull response:

```
{
  "status": 2000,
  "quote": Quote POD
}
```

Resources:

- [Quote POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Invalid account id.	Wrong account id specified.
401	4010	Not authorized.	You are not authorized, contact support.
400	4402	Invalid data provided.	Invalid quote data provided in the request, see details in the response.
500	5000	Internal error.	Internal error occurred, contact support.

4.5.4 Resend Quote

Resend a specific quote.

Endpoint URL:

Method Type	Url
GET	api.brighte/rest/vendor/quote/{id}/resend

Parameters:

Name	Value	Description
id	string	Id of the quote.

Sample request:

HTTP

```
POST /rest/vendor/v1/quote/{id}/resend HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful response:

```
{
  "status": 2000,
  "quote": Quote POD
}
```

Resources:

- [Quote POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Quote already applied.	Cant resend this quote, an application has already been filed with Brighte.
400	4402	The quote is invalid.	There is a problem with the quote, please create a new one.
400	4402	The quote is expired.	The quote is expired already, please create a new one.
400	4402	Invalid quote ID.	Please specify a quote id.
404	4041	Invalid quote ID.	Quote not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid quote ID.	You are not authorized to access this resource.

4.5.5 Delete Quote

Cancel/delete a specific quote.

Endpoint URL:

Method Type	Url
DELETE	api.brighte/rest/vendor/quote/{id}

Parameters:

Name	Value	Description
id	string	Id of the quote.

Sample request:

HTTP

```
DELETE /rest/vendor/v1/quote/{id} HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful response:

```
{
  "status": 2000,
  "quote": Quote POD
}
```

Resources:

- [Quote POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Quote not active.	Can't cancel a quote that is expired or inactive.
400	4402	Invalid quote ID.	Please specify a quote id.
404	4041	Invalid quote ID.	Quote not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid quote ID.	You are not authorized to access this resource.

4.6 Users

4.6.1 Get Users

The results are ordered by creation date. There is a maximum limit of 500 records per call and the next result set can be retrieved by utilizing the *pageNumber* parameter and the *hasMoreRecords* return field.

Endpoint URL:

Method Type	Url
GET	portal.brighte.com.au/rest/vendor/v1/users

Parameters:

Name	Value	Description
accountId	string	Account to retrieve the users for.
startDate	date	(optional, default beginning) Start Date
endDate	date	(optional, default today) End Date
pageNumber	integer	(optional, default 1) Page Number

Sample request:

HTTP

```
GET /rest/vendor/v1/users?accountId=xxx&startDate=2018-01-01&endDate=2018-01-31&
    ↪pageNumber=1 HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful Sample Response:

```
{
  "status": 2000,
  "users": [
    User POD
  ],
  "hasMoreRecords": boolean
}
```

Resources:

- [User POD](#)

Error codes:

Code	Status	Message	Reason
401	4010	Invalid account id.	Please specify a valid account id.

4.6.2 Get User

Return information about a specific user.

Endpoint URL:

Method Type	Url
GET	portal.brighte.com.au/rest/vendor/v1/user/{user_id}

Parameters:

Name	Value	Description
userId	string	User to retrieve.

Sample request:

HTTP

```
GET /rest/vendor/v1/user/{user_id} HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
```

(continues on next page)

(continued from previous page)

```
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful Sample Response:

```
{
  "status": 2000,
  "user": "User POD"
}
```

Resources:

- [User POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Invalid user ID.	Please specify a user id.
404	4041	Invalid user ID.	User not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid user ID.	You are not authorized to access this resource.

4.6.3 Create User

Create a user with a specific account and user role.

Endpoint URL:

Method Type	Url
POST	portal.brighte.com.au/rest/vendor/v1/user/add

Parameters:

Name	Value	Description
account_id	string	Account ID.
role	enum	See userRole.
first_name	string	Users first name.
last_name	string	Users last name.
email	string	Unique email of the user.
mobile	string	Mobile phone number without country code.

Sample request:**HTTP**

```
POST /rest/vendor/v1/user/add HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
```

(continues on next page)

(continued from previous page)

```
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache

{
  "role": "team+member",
  "first_name": "Testfn",
  "last_name": "TestLn",
  "email": "first.last1536623737@brighte.com.au",
  "mobile": "0411085823",
  "account_id": "E81"
}
```

Successful Sample Response:

```
{
  "status": 2000,
  "user": "User POD"
}
```

Resources:

- [User POD](#)

Error codes:

Code	Status	Message	Reason
401	4010	Invalid account id.	Please specify a valid account id.
401	4010	Not authorized.	You are not authorized to access this resource.
400	4402	Invalid role.	Please specify a valid role.
400	4402	Invalid data provided.	See <i>details</i> in the response.

4.6.4 Update User

Update a user.

Endpoint URL:

Method Type	Url
PUT	portal.brighte.com.au/rest/vendor/v1/user/{userId}

Parameters:

Name	Value	Description
userId	string	User to update.
role	enum	See <code>userRole</code> .
first_name	string	Users first name.
last_name	string	Users last name.
email	string	Unique email of the user.
mobile	string	Mobile phone number without country code.

Sample request:

HTTP

```
PUT /rest/vendor/v1/user/{userId} HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache

{
  "first_name": "johnny",
  "last_name": "bravo"
}
```

Successful Sample Response:

```
{
  "status": 2000,
  "user": "User POD"
}
```

Resources:

- [User POD](#)

Error codes:

Code	Status	Message	Reason
401	4010	Not authorized.	You are not authorized to access this resource.
400	4402	Invalid role.	Please specify a valid role.
400	4402	Invalid data provided.	See <i>details</i> in the response.
403	4403	Action not allowed.	Contact support for more explanation.

4.6.5 Delete User

Delete a specific user.

Endpoint URL:

Method Type	Url
DELETE	portal.brighte.com.au/rest/vendor/v1/user/{userId}

Parameters:

Name	Value	Description
userId	string	User to delete.

Sample request:

HTTP


```
DELETE /rest/vendor/v1/user/{userId} HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful Sample Response:

```
{
  "status": "2000"
}
```

Error codes:

Code	Status	Message	Reason
400	4402	Invalid user ID.	Please specify a user id.
404	4041	Invalid user ID.	User not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid user ID.	You are not authorized to access this resource.

4.6.6 Generate User Api Credentials

Generate API-Id and API-Secret for a specific user.

Endpoint URL:

Method Type	Url
POST	portal.brighte.com.au/rest/vendor/v1/user/{userId}/generate-api-credentials

Parameters:

Name	Value	Description
userId	string	User to generate the api credentials for.
account_id	string	Credential to access this account.

Sample request:

HTTP

```
POST /rest/vendor/v1/user/{userId}/generate-api-credentials HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache

{
  "accountId": "E81"
}
```

Successful Sample Response:

```
{
  "status": 2000,
  "userKey": "UserKey POD"
}
```

Resources:

- [UserKey POD](#)

Error codes:

Code	Status	Message	Reason
401	4010	Not authorized.	You are not authorized to access this resource.
400	4402	Invalid user ID.	Please specify a user id.
404	4041	Invalid user ID.	User not found.
500	5000	Internal error.	There was an error with this request, contact support.
401	4010	Invalid user ID.	You are not authorized to access this resource.
400	4402	Invalid data provided.	See <i>details</i> in the response.

4.7 Content

4.7.1 Get Content

Retrieve documents such as privacy policy, terms and condition or direct debit agreement.

Endpoint URL:

Method Type	Url
GET	portal.brighte.com.au/rest/vendor/v1/content/{contentType}

Parameters:

Name	Value	Description
contentType	enum	Type of document.

Sample request:

HTTP

```
GET /rest/vendor/v1/content/{contentType} HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful Sample Response:

```
{
  "status": 2000,
  "content": Content POD
}
```

Resources:

- [Content POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Invalid input.	Invalid input parameters.

4.8 Product Categories

4.8.1 Get Product Categories

Return all possible values for product category.

Endpoint URL:

Method Type	Url
GET	portal.brighte.com.au/rest/vendor/v1/product-categories

Parameters:

No parameters

Sample request:

HTTP

```
GET /rest/vendor/v1/product-categories HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache
```

Successful Sample Response:

```
{
  "status": 2000,
  "productCategories": [
    ProductCategory POD
  ]
}
```

Resources:

- [ProductCategory POD](#)

Error codes:

No error codes

4.9 Calculate Repayment Amount

4.9.1 Calculate Repayment Amount

Calculate the repayment amount of the loan.

Endpoint URL:

Method Type	Url
POST	portal.brighte.com.au/rest/v1/vendor/calculate-repayment-amount

Parameters:

Name	Value	Description
totalPurchaseAmount	currency	Total amount of the loan.
depositAmount	currency	The deposit of the loan, will be subtracted from the total amount.
repaymentTerm	integer	The repayment term in months.
includeFees	boolean	Whether to include the account establishment fee for first time customers.

Sample request:

HTTP

```
POST /rest/vendor/v1/calculate-repayment-amount HTTP/1.1
Host: brighte.localhost
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Accept: application/json
Cache-Control: no-cache

{
  "totalPurchaseAmount" : 15000,
  "depositAmount": 1000,
  "repaymentTerm" : 48,
  "includeFees" : 1
}
```

Successfull response:

```
{
  "status": 2000,
  "repaymentAmount": RepaymentAmount POD
}
```

Resources:

- [RepaymentAmount POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Invalid input.	Invalid input parameters.

4.10 Callout Requests

4.10.1 Create Callout Request

Initiate a callout request. If successful, the call will be made at the earliest available date and time, described in the returned *call_datetime* field.

Endpoint URL:

Method Type	Url
POST	portal.brighte.com.au/rest/vendor/v1/callout-requests

Parameters:

Name	Value	Description
first_name	string	First name of the client.
last_name	string	Last name of the client.
phone	string	Phone number of the client.
email	string	Email of the client.
total_purchase_amount	currency	Proposed financed amount.
deposit_amount	currency	Proposed deposit amount.
repayment_term	integer	Repayment term in months.
product_category	string	Category of the product in this application.
product_description	string	Description of the financed items.

Sample request:

HTTP

```
POST /rest/vendor/v1/callout-requests HTTP/1.1
Host: portal.brighte.com.au
Accept: application/json
Content-Type: application/json
X-Session-Key: xxx
X-Session-Token: xxx
Cache-Control: no-cache

{
  "account_id" : 3,
  "first_name" : "John",
  "last_name" : "Smith",
  "phone" : "0434090090",
  "email" : "email@gmail.com",
  "total_purchase_amount" : 15000,
  "deposit_amount" : 5000,
```

(continues on next page)

(continued from previous page)

```
"repayment_term" : 48,  
"product_category" : "Storage Battery(s) ",  
"product_description" : "Excellent Battery"  
}
```

Successful Sample Response:

```
{  
  "status": 2000,  
  "callout": Callout POD  
}
```

Resources:

- [Callout POD](#)

Error codes:

Code	Status	Message	Reason
400	4402	Invalid input.	Invalid input parameters.
401	4010	Not authorized.	You are not authorized.
500	5000	Internal error.	Internal error occurred, contact support.

CHAPTER 5

API Data Types

Here is a complete list of data types used by Brighte API.

5.1 Simple Data Types

Data Type	Description
times-tamp	String representation of date and time value, up to seconds, in ISO8601 e.g. '2012-10-01T00:00:00+10:00'
date	String representation of date value, in YYYY-MM-DD format. e.g. '2012-10-01'
currency	64-bit signed integer in cents (i.e. range from -9223 trillion to 9223 trillion). A leading minus sign indicates a negative number. No leading plus sign is used for positive number. e.g. \$238.50 will be presented as 23850 -\$37.95 will be presented as -3795
percentage	32-bit signed integer (i.e. range from -2147 million to 2147 million). Assumption: no more than 2 decimal places is used. e.g. 5.95% is presented as 595. -2% is presented as -200. 0.89% is presented as 89.
string	UTF-8 encoded string.
file	String representation of file (base64 encoded).
integer	64-bit signed integer (i.e. range from -9223 trillion to 9223 trillion). A leading minus sign indicates a negative number. No leading plus sign is used for positive number.
boolean	integer 1 or integer 0

5.2 Enums

Enum	Possible Values
applicationStatus	APPROVED C.APPROV DECLINED PAID QUOTE REFER WITHDRAW
applicationDocument-Type	INVOICE STC CCA
calloutStatus	OPEN FULLFILLED CLOSED
contentType	DIRECT_DEBIT_AGREEMENT PRIVACY_POLICY TNC
gender	FEMALE MALE
identificationType	DRIVER_LICENCE MEDICARE_CARD PASSPORT
maritalStatus	DEFACTO MARRIED SINGLE
medicareColour	BLUE GREEN YELLOW
paymentMethod	BANK_ACCOUNT VISA MASTERCARD
pensionerOrVeteranType	AGED_PENSION RETURNED_SERVICEMEN_PENSION SELF_FUNDED_RETIREE
personTitle	DR MISS MR MRS MS
quoteStatus	COMPLETED SENT
stateCode	ACT NSW NT QLD SA TAS VIC WA
userRole	ADMINISTRATOR TEAM MEMBER

5.3 Complex Data Types (aka POD objects)

Complex data types are implemented as POD objects.

5.3.1 Application

JSON template for **Application** POD:

```
{
  "id": string,
  "applicants": Applicant POD[],
  "team_member": string,
  "financed_amount": currency,
  "repayment_term": integer,
  "product_category": string,
  "product_description": string,
  "status": enum<applicationStatus>,
  "created": timestamp
}
```

Resources:

- [Applicant POD](#)

Name	Value	Description
id	string	This is the unique id.
applicants	Applicant POD[]	Primary and joint applicants.
team_member	string	Name of the agent submitted the application: <first_name> <last_name>
financed_amount	currency	How much to finance with this application.
repayment_term	integer	In months.
product_category	string	Category of the product in this application.
product_description	string	Description of the financed items.
status	enum	See applicationStatus.
created	timestamp	Time of creation.

5.3.2 Applicant

JSON template for **Applicant** POD:

```
{
  "id": string,
  "name": string,
  "email": string,
  "mobile": string,
  "dob": date,
  "joint_applicant": boolean
}
```

Name	Value	Description
id	string	This is the unique id.
name	string	Full name of the applicant.
email	string	Email of the applicant.
mobile	string	Mobile phone number of the applicant.
dob	date	Date of birth of the applicant.
joint_applicant	boolean	Flag indicating if this is the joint applicant or the primary applicant.

5.3.3 ApplicationDocument

JSON template for **ApplicationDocument** POD:

```
{
  "type": enum<applicationDocumentType>,
  "filename": string,
}
```

Name	Value	Description
type	enum	See applicationDocumentType.
filename	string	Name of the file.

5.3.4 Quote

JSON template for **Quote** POD:

```
{
  "id": string,
  "applicant_email": string,
  "total_purchase_amount": currency,
  "deposit_amount": currency,
  "repayment_term": integer,
  "product_category": string,
  "product_description": string,
  "created": timestamp,
  "team_member": string,
  "applicant": string,
  "status": enum<quoteStatus>
}
```

Name	Value	Description
applicant_email	string	Email of the applicant and the target where the quote is to be sent.
total_purchase_amount	currency	Proposed financed amount.
deposit_amount	currency	Proposed deposit amount.
repayment_term	integer	Repayment term in months.
product_category	string	Category of the product in this application.
product_description	string	Description of the financed items.
created	timestamp	Time the quote was created.
id	string	Unique ID.
team_member	string	Name of the agent submitted the application: <first_name> <last_name>
applicant	string	Name of applicant: <first_name> <last_name>
status	enum	See quoteStatus.

5.3.5 User

JSON template for **User** POD:

```
{
  "id": string,
  "role": enum<userRole>,
  "firstName": string,
  "lastName": string,
  "email": string,
  "mobile": string
}
```

Name	Value	Description
id	string	Unique ID.
role	enum	See userRole.
first_name	string	Users first name.
last_name	string	Users last name.
email	string	Unique email of the user.
mobile	string	Mobile phone number without country code.

5.3.6 Account

JSON template for **Account** POD:

```
{
  "id": integer,
  "entity_legal_name": string,
  "entity_trading_name": string,
}
```

Name	Value	Description
id	string	Unique ID.
entity_legal_name	string	Trading name.
entity_trading_name	string	Legal name.

5.3.7 Content

JSON template for **Content** POD:

```
{
  "content_type": enum<contentType>,
  "version": timestamp,
  "content": string,
}
```

Name	Value	Description
content_type	enum<contentType>	Unique ID.
version	timestamp	Trading name.
content	string	Legal name.

5.3.8 ProductCategory

JSON template for **ProductCategory** POD:

```
{
  "name": string,
  "slug": string,
  "icon": string
}
```

Name	Value	Description
name	string	Legal name.
slug	string	Slugged representation of the name (url safe).
icon	string	Product icon url.

5.3.9 RepaymentAmount

JSON template for **ProductCategory** POD:

```
{
  "repaymentAmount": currency,
  "instalments": integer
}
```

Name	Value	Description
repaymentAmount	currency	The amount of each instalment.
instalments	integer	How many instalments over the desired repayment term.

5.3.10 Callout

JSON template for **Callout** POD:

```
{
  "id": string,
  "first_name": string,
  "last_name": string,
  "phone": string,
  "email": string,
  "total_purchase_amount": currency,
  "deposit_amount": currency,
  "repayment_term": integer,
  "product_category": string,
  "product_description": string,
  "call_datetime": timestamp,
  "status": enum<calloutStatus>,
  "created": timestamp,
  "modified": timestamp,
  "full_name": string,
  "financed_amount": currency
}
```

Name	Value	Description
id	string	Unique ID.
first_name	string	First name of the client.
last_name	string	Last name of the client.
phone	string	Phone number of the client.
email	string	Email of the client.
total_purchase_amount	currency	Proposed financed amount.
deposit_amount	currency	Proposed deposit amount.
repayment_term	integer	Repayment term in months.
product_category	string	Category of the product in this application.
product_description	string	Description of the financed items.
call_datetime	timestamp	Time of the call.
status	enum	See calloutStatus.
created	timestamp	Time the quote was created.
modified	timestamp	Time the quote was created.
applicant_email	string	Email of the applicant and the target where the quote is to be sent.
full_name	string	Full name of the client: <first_name> <last_name>
financed_amount	currency	Proposed financed amount.

5.3.11 UserKey

JSON template for **UserKey** POD:

```
{
  "apiId": string,
  "apiSecret": string
}
```

Name	Value	Description
apiId	string	Api ID.
apiSecret	string	Api Secret.