

HTTP API Documentation

Version 1.1

135 Tech Labs

Contents

1	HTTP API	3
1.1	Access Token API	3
1.1.1	Parameters	4
1.1.2	Response	4
1.2	Reward List API	5
1.2.1	Parameters	6
1.2.2	Response	9
1.3	Reward Redeem API	10
1.3.1	Parameters	11
1.3.2	Response	11
1.4	All Rewards Redeemed API	12
1.4.1	Response	12
1.5	Hangout Points	13

List of Tables

1.1	Parameters required for a Token request API call	4
1.2	Compulsory parameters for a List API call	5
1.3	Optional parameters for a List API call	6
1.4	Compulsory parameters for a Redeem API call	10
1.5	Optional parameters for a Redeem API call	10
1.6	Compulsory parameters for a Redeem API call	12

Chapter 1

HTTP API

The HTTP API provides an independent and uniform interface for third-party applications to interact with the Hangoutt platform. Through the use of universally accessible HTTP endpoints, the API implements a simple and stateless functionality for disbursement of rewards which are available with Hangoutt. Due to the universality of the underlying HTTP, this API will be accessible across all mobile OS flavors, and across different development environments.

The base URL for all HTTP API endpoints is:

```
https://api.hangoutt.com
```

We use OAuth2 for authentication and authorization, and as such all API endpoints (except for the token request endpoint) will require a valid access token.

1.1 Access Token API

Every API call requires a valid access token for successful invocation. An access token can be obtained by making a HTTP POST request to the URL:

```
https://api.hangoutt.com/token.php
```

Key	Value
grant_type	client_credentials
client_id	<your_client_id>
client_secret	<your_client_secret>

Table 1.1: Parameters required for a Token request API call

1.1.1 Parameters

The POST request expects three parameters as part of the request body, which are listed in Table 1.1. The `client_id` and `client_secret` for your application can be retrieved from the developers portal.

1.1.2 Response

The response to the token API endpoint is in the form of a JSON string. It includes the access token, its expiry information, the token type and its scope. An example response is shown in the listing below.

```
{
  "access_token": "<your_access_token>",
  "expires_in": 3600,
  "token_type": "Bearer",
  "scope": "default"
}
```

As indicated in the JSON response, an access token is valid for the duration of an hour. A new request for an access token has to be made upon expiry. No refresh tokens are provided.

When any HTTP API endpoint is invoked with an expired access token, a response code of 401 is returned, along with an explanation in the WWW-Authenticate HTTP header. The HTTP response would be as shown below.

```
HTTP/1.1 401 Unauthorized
WWW-Authenticate: error="expired_token"
```

Key	Value
access_token	<your_access_token>
points	<number_of_points>

Table 1.2: Compulsory parameters for a List API call

The third party application thus need not keep track of the lifetime of an access token. All it has to do, is to look out for the unauthorized error and request for a new access token when it appears.

1.2 Reward List API

The List API call is used to fetch a list of available rewards for the user. This API endpoint can be called at any time to retrieve a list of rewards. The List API endpoint is at:

```
https://api.hangoutt.com/rewards/list.php
```

The List API endpoint expects a HTTP POST request, with parameters passed as part of the request body. The List API expects a few compulsory parameters which have to be **always** passed on an invocation of the endpoint. These parameters are listed in Table 1.2.

In addition to the compulsory parameters, a number of optional parameters can be passed to the List API endpoint, thereby controlling both categorization and ranking of rewards on the server side. The optional parameters are listed in Table 1.3. The expected values for each of the parameters are explained in Section 1.2.1. The response format of the List API is explained in Section 1.2.2.

Key	Value
location	<location_json>
limit	<number_of_rewards_returned>
exclude	<exclude_json>
user	<user_json>
time	<epoch>
labels	<labels_json>

Table 1.3: Optional parameters for a List API call

1.2.1 Parameters

Points

The points parameter is used to indicate how many points the end user currently has accumulated with the SDK. This is used to check if the user is eligible to receive a given reward. It is to be noted that this is the number of **Hangoutt Points** that the user has accumulated. Please refer to Section 1.5 to understand how Hangoutt points are computed.

Location

The location parameter expects a well formed JSON string as the value, in the following format:

```
{
  "country_code": "<your_country_code>",
  "country": "<your_country>",
  "state": "<your_state>",
  "city": "<your_city>"
}
```

If no value is specified for the location parameter, the API server performs a lookup based on the client IP to determine the location. Most third-party applications do not have a way of determining the exact location, and as such can choose to not specify any value for this parameter.

It should be noted that if a value is provided for the location attribute, but no country code is provided as part of the value, then the server reverts to determining the location based on client IP.

Limit

The limit parameter can be used to limit the number of rewards returned as part of the List API response. There is an upper limit of five (5) rewards which are returned as part of any response to the List API call. A minimum of one (1) reward will always be returned, so a valid value for the limit parameter is in the range of [1-5].

Exclude

The exclude parameter can be used to exclude rewards which have already been seen from consideration by the List API . As part of the List API response, reward ID's are returned which identify each reward in the Hangoutt platform uniquely. The exclude parameter takes a list of such reward ID's as its value, and directs the List API to return a response which does not contain these reward ID's. A valid value for the exclude parameter is a well-formatted JSON as shown below:

```
[
  "<reward_ID_1>",
  "<reward_ID_2>",
  "...",
  "<reward_ID_n>"
]
```

User

The user parameter can be used to specify characteristics of the end user (not the developer) requesting the rewards. This is useful to the Hangoutt platform in trying to determine the most suitable rewards for each specific end user. Such details are hard to come by for third-party applications unless they maintain their own user database, or use third-party authentication (e.g. Facebook) while requesting authorization to access specific user details. A valid value for the user parameter is in the form of a well-formatted JSON as shown below:


```
{
  "email": "<end_user_email>",
  "gender": "<end_user_gender>",
  "age_group": [
    "<start_age>",
    "<end_age>"
  ]
}
```

The gender key can take one of the following values: male, female, all.

Time

The local time can be specified by passing the current unix epoch time with this parameter. The List API uses this information to rank rewards based on time of the day.

Labels

The labels parameter is used to identify and classify the instance of time when a call to the List API is made. We provide statistics based on labels for developers, which will be useful to glean information on which calls to List API are fetching them the highest redemptions, how many are converting to sales etc. A valid value for the labels parameter is a well-formatted JSON as shown below:

```
[
  "<label_1>",
  "<label_2>",
  "...",
  "<label_n>"
]
```

For example, a possible labels array could look like this:

```
[
  "Engagement time 10 minutes",
  "Reason In-App Purchase"
]
```

1.2.2 Response

The response to a valid invocation of the List API call is in the form of a well-formatted JSON string as shown below:

```
{
  "valid": 1,
  "rewards": {
    "brand11403691635ban1412178869": {
      "image_url": "<reward_image_url>",
      "points": "200"
    },
    "brand11403691635ban1416389303": {
      "image_url": "<reward_image_url>",
      "points": "250"
    },
    "brand21403778127hyd1416477417": {
      "image_url": "<reward_image_url>",
      "points": "150"
    },
    "brand31403779347hyd1406551435": {
      "image_url": "<reward_image_url>",
      "points": "30"
    }
  }
}
```

In the response, a valid value of 1 means that the API processed the request successfully. All unsuccessful invocations at the API level have the valid field set to 0. If there is an accompanying description of the error, it is set on the **message** key in the response (only part of unsuccessful invocations' responses).

The rewards map contains information about the reward ID's, their associated images as well as the cost in terms of the number of points. The reward ID's listed in the sample response are just examples, and do not represent actual rewards.

It should be noted that the exclude parameter discussed in Section 1.2.1 requires a list of such reward ID's as its value.

Key	Value
access_token	<your_access_token>
points	<number_of_points>
id	<reward_id>
user	<user_json>

Table 1.4: Compulsory parameters for a Redeem API call

Key	Value
location	<location_json>
time	<epoch>
labels	<labels_json>

Table 1.5: Optional parameters for a Redeem API call

1.3 Reward Redeem API

The Redeem API is used to redeem a selected reward, for which the user is eligible for after having earned the required number of points. The Redeem API endpoint is at:

```
https://api.hangoutt.com/rewards/redeem.php
```

The Redeem API endpoint expects a HTTP POST request, with parameters passed as part of the request body. The Redeem API expects a few compulsory parameters which have to be **always** passed on an invocation of the endpoint. These parameters are listed in Table 1.4.

In addition to the compulsory parameters, a number of optional parameters can be passed to the Redeem API endpoint. The optional parameters are listed in Table 1.5. The expected values for each of the parameters are explained in Section 1.3.1. The response format of the List API is explained in Section 1.3.2.

1.3.1 Parameters

In this section, we explain the meaning and value formats for most parameters which are passed to the Redeem API. The points, location, time and labels parameters are completely similar to how they are specified for the List API, and we refer the reader to Section 1.2.1 for an explanation for these three parameters.

ID

The id parameter specifies the chosen reward by indicating its unique id in the Hangout platform. This id is communicated to the third-party application in the List API Call's response, as indicated in Section 1.2.2. For the sample response shown in Section 1.2.2, the id parameter would be set to the value

```
"brand11403691635ban1412178869"
```

if the first reward in the list were chosen for redemption by the end user.

User

The user parameter is used to identify the end user who wishes to redeem the reward. A valid value for the user parameter is in the form of a well-formatted JSON as shown below:

```
{
  "email": "<end_user_email>"
}
```

1.3.2 Response

The response to a valid invocation of the Redeem API call is in the form of a well-formatted JSON string as shown below:

```
{
  "valid": 1,
  "deducted": 30
}
```

Key	Value
access_token	<your_access_token>
email	<end_user_email>

Table 1.6: Compulsory parameters for a Redeem API call

In the response, a valid value of 1 means that the API processed the request successfully. All unsuccessful invocations at the API level have the valid field set to 0. If there is an accompanying description of the error, it is set on the **message** key in the response (only part of unsuccessful invocations' responses).

The **deducted** key in the response indicates how many points should be deducted from the user's accumulated score for redeeming the chosen reward. Maintenance of the accumulated score is the responsibility of the third party application, so this API response clearly specifies how many points need to be deducted from the accumulated score.

1.4 All Rewards Redeemed API

The All Rewards Redeemed API is used to fetch all redeemed rewards for a particular user. The All Rewards Redeemed API endpoint is at:

```
https://api.hangoutt.com/rewards/all_redeemed.php
```

The All Rewards Redeemed API endpoint expects a HTTP POST request, with parameters passed as part of the request body. The API expects two compulsory parameters which have to be **always** passed on an invocation of the endpoint. These parameters are listed in Table 1.6.

1.4.1 Response

The response to a valid invocation of the All Rewards Redeemed API call is in the form of a well-formatted JSON string. The response consists of the details of all the rewards redeemed by the user. A valid response JSON string is as shown below:

```
{
  "valid": 1,
  "rewards": [
    {
      "title": "<reward_title>"
      "description": "<reward_description>",
      "t_and_c": "<reward_terms_and_conditions>",
      "coupon_code": "<coupon_code>",
      "image_url": "<reward_image_url>",
      "image_avg_color": "<reward_image_avg_color>",
      "link": "<reward_target_link>"
    }
  ]
}
```

In the response, a valid value of 1 means that the API processed the request successfully. All unsuccessful invocations at the API level have the valid field set to 0. If there is an accompanying description of the error, it is set on the **message** key in the response (only part of unsuccessful invocations' responses).

The **rewards** array contains details of all the redeemed rewards. The title, description and coupon code of the particular reward are associated with the corresponding keys. If the reward has any associated terms and conditions, they are set on the **t_and_c** key. The reward image (or reward creative) can be found at the URL pointed to by the **image_url** key.

A suitable background color value associated with the reward image is available with the **image_avg_color** key. This value is in the form of RGB values, which can be used to paint a suitable background color on which the reward image can be overlaid. The reward itself is available at a target link pointed to by the **link** key of the reward.

1.5 Hangoutt Points

Every third-party application has its own way of scoring, with scores not being directly comparable across applications in terms of either effort rendered or time spent. The Hangoutt reward platform, thus, had to define a universal scale of scoring so that consistency in number of points earned is maintained across applications.

We use a very simple formula to compute the user's score:

```
10 minutes of effort = 100 Hangoutt points.
```

In other words, we score based on the amount of time spent by a user actively engaging with the application. An active engagement duration of 10 minutes will fetch the user an average of 100 Hangoutt points. This is the benchmark that we follow in our applications, and recommend that every third-party application reporting scores to Hangoutt reward platform should follow the same benchmark.