
Connect Javascript SDK

Release 19.0.5

Ingram Micro Inc

Dec 30, 2019

CONTENTS

1	Introduction	1
2	Table of contents	3
2.1	Getting started	3
2.2	API Reference	4
3	Indices and tables	17
4	License	19
	Index	21

INTRODUCTION

CloudBlue Connect is a supply automation platform that manages your products and services, contracts, ordering and fulfillment, usage and subscriptions.

It supports any product, from physical goods to cloud products, as well as any channel, including your direct and indirect sales channels and internal procurement.

With its flexible APIs, it can connect to any commerce platform.

Vendors can leverage CloudBlue Connect to:

- Reduce the total cost of ownership for homegrown technology supporting their indirect channel
- Standardize integrations with partners
- Increase efficiencies and minimize redundancies by bridging their direct and indirect sales channels

Service providers can use CloudBlue Connect to:

- Define, manage and distribute any type of product (omni-product) through any channel (omni-channel)
- Transform perpetual licensed products into a subscription model
- Onboard new products into their portfolio quickly to build and deliver unique solutions to end customers

The Cloud Blue Connect Javascript SDK helps developers to integrate your commerce platform with Cloud Blue Connect.

TABLE OF CONTENTS

2.1 Getting started

2.1.1 Requirements

The Cloud Blue Connect Javascript SDK depends on the following 3rd party packages:

- `lodash` version 4.17.15
- `node-fetch` version 2.6.0

2.1.2 Installation

The current stable release of the Connect Javascript SDK is 19.0.5.

To install the Connect Javascript SDK at the command prompt enter:

```
$ npm install @cloudblueconnect/connect-javascript-sdk --save
```

2.1.3 Usage

The `connect-javascript-sdk` allow you to consume the Connect public API directly or using specialized methods to achieve tasks related to a specific workflow.

High-level Usage

```
const { ConnectClient, Fulfillment } = require('@cloudblueconnect/connect-javascript-  
  ↳ sdk');  
const client = new ConnectClient('<api_endpoint>', '<api_key>');  
const fulfillment = new Fulfillment(client);  
  
const response = await fulfillment.approveRequestWithTemplate('<request_id>', '  
  ↳ <template_id>');
```

Low-level Usage

You can access the API endpoints directly through the `ConnectClient`.

The `ConnectClient` groups operations by the resource they access.

Create an instance of the `ConnectClient` class:

```
const { ConnectClient } = require('@cloudblueconnect/connect-javascript-sdk');  
  
const client = new ConnectClient('<api_endpoint>', '<api_key>');
```

For example to invoke the list operation for the `Product` resource:

```
client.products.list()  
  .then(products => { console.log(products) });
```

Or if you prefer the `async/await` syntax:

```
const products = await client.products.list();  
console.log(products);
```

2.2 API Reference

2.2.1 Client

class `ConnectClient` (*endpoint*, *apiKey*, *adapter*)

`ConnectClient` provides access to resource endpoints of the Cloud Blue Connect Platform.

Creates a new instance of the Cloud Blue `ConnectClient` class. Configures the `beforeRequest` hooks to inject the API key in the Authorization header and prefixes the url with the API's base url.

Arguments

- **endpoint** (*string*) – The base URL for API access.
- **apiKey** (*string*) – The API key to authenticate with Connect.
- **adapter** (*AbstractHttpAdapter*) – An optional adapter.

Returns `ConnectClient` – An instance of the `ConnectClient` class.

`ConnectClient.accounts`

type: `AccountService`

Returns an instance of the `AccountService` class to access the `Account` resource endpoint.

`ConnectClient.actions`

type: `ActionService`

Returns an instance of the `ActionService` class to access the product `Action` resource endpoint.

`ConnectClient.conversations`

type: `ConversationService`

Returns an instance of the `ConversationService` class to access the `Conversation` resource endpoint.

`ConnectClient.hubs`

type: `HubService`

Returns an instance of the `HubService` class to access the `Hub` resource endpoint.

`ConnectClient.products`

type: ProductService

Returns an instance of the *ProductService* class to access the `Product` resource endpoint.

`ConnectClient.requests`

type: RequestService

Returns an instance of the *RequestService* class to access the `Request` resource endpoint.

`ConnectClient.tierAccounts`

type: TierAccountService

Returns an instance of the *TierAccountService* class to access the `TierAccount` resource endpoint.

`ConnectClient.tierConfigRequests`

type: TierConfigRequestService

Returns an instance of the *TierConfigRequestService* class to access the `TierConfigurationRequest` resource endpoint.

2.2.2 Operations

class `fulfillment.Fulfillment` (*client*)

The `Fulfillment` class exposes specialized methods to help developers to achieve common use cases for the fulfillment workflow.

Creates an instance of the `Fulfillment` class.

Arguments

- **client** (`ConnectClient`) – An instance of the `ConnectClient` class.

Returns `Fulfillment` – An instance of the `Fulfillment` class.

`fulfillment.Fulfillment.approveRequest` (*id*, *request*)

Updates the `Request` and set its status to 'approved'.

Arguments

- **id** (*string*) – The unique identifier of the `Request` object.
- **request** (*Object*) – The request body.

Returns `Object` – The updated `Request` object.

Examples:

```
// request body using a template id
{
  template_id: 'TL-827-840-476'
}
```

```
// request body using an activation tile
{
  activation_tile: '<rendered text>'
}
```

`fulfillment.Fulfillment.approveRequestWithTemplate` (*id*, *templateId*)

Updates the `Request` and set its status to 'approved' using an activation template.

Arguments

- **id** (*string*) – The unique identifier of the Request object.
- **templateId** (*string*) – The template id to use for Request approval.

Returns Object – The updated Request object.

`fulfillment.Fulfillment.approveTierConfigRequest (id, request)`
Updates the TierConfigurationRequest and set its status to 'approved'.

Arguments

- **id** (*string*) – The unique identifier of the Request object.
- **request** (*Object*) – The request body.

Returns Object – The updated Request object.

Examples:

```
// request body using a template id
{
  template: {
    id: 'TL-827-840-476'
  }
}
```

`fulfillment.Fulfillment.approveTierConfigRequestWithTemplate (id, request, templateId)`
Updates the TierConfigurationRequest and set its status to 'approved' using an template.

Arguments

- **id** (*string*) – The unique identifier of the TierConfigurationRequest object.
- **templateId** (*string*) – The template id to use for TierConfigurationRequest approval.

Returns Object – The updated Request object.

`fulfillment.Fulfillment.createRequest (request)`
Creates a new Request object.

Arguments

- **request** (*Object*) – The Request object to create.

`fulfillment.Fulfillment.createTierConfigRequest (request)`
Creates a new TierConfigurationRequest object.

Arguments

- **request** (*Object*) – The TierConfigurationRequest object to create.

`fulfillment.Fulfillment.failRequest (id, reason)`
Changes the status of the Request object to 'fail'.

Arguments

- **id** (*string*) – The unique identifier of the Request object.
- **reason** (*string*) – The reason for which the Request has been failed.

Returns Object – The updated Request object.

`fulfillment.Fulfillment.failTierConfigRequest (id, reason)`
Changes the status of the TierConfigurationRequest object to 'fail'.

Arguments

- **id** (*string*) – The unique identifier of the TierConfigurationRequest object.
- **reason** (*string*) – The reason for which the TierConfigurationRequest has been failed.

Returns Object – The updated Request object.

fulfillment.Fulfillment.**inquireRequest** (*id, request, params, note*)

Updates the Request object to inquire the provider for ordering parameter. It updates the Parameter object **value_error** attribute and set the status of the Request to 'inquire'.

Arguments

- **id** (*string*) – The unique identifier of the Request object.
- **request** (*Object*) – The request body.
- **params** (*Array*) – An array of Parameter objects to update.
- **note** (*string*) – An optional 'note' attribute for the Request.

Returns Object – The updated Request object.

Examples:

```
// example of the params argument.
[
  {
    id: 'param_a', // id is required
    value_error: 'This address is already used. Try another.'
  }
]
```

```
// request body using a template id
{
  template_id: 'TL-827-840-476'
}
```

```
// request body using an activation tile
{
  activation_tile: '<rendered text>'
}
```

fulfillment.Fulfillment.**inquireRequestWithTemplate** (*id, templateId, params, note*)

Updates the Request object to inquire the provider for ordering parameter using an activation template. It updates the Parameter object **value_error** attribute and set the status of the Request to 'inquire'.

Arguments

- **id** (*string*) – The unique identifier of the Request object.
- **templateId** (*string*) – The unique identifier of the Template object.
- **params** (*Array*) – An array of Parameter objects to update.
- **note** (*string*) – An optional 'note' attribute for the Request.

Returns Object – The updated Request object.

fulfillment.Fulfillment.**listRequests** (*filters, orderBy, limit=100, offset=0*)

Returns a list of at most **limit** Request objects that match the provided filters. If no filter is passed, a page of Request in 'pending' status is returned.

See the *RequestService.list method* for further information about the filters object.

Arguments

- **filters** (*Object*) – A filter object.
- **orderBy** (*string/Array*) – The ordering criteria.
- **limit** (*number*) – The maximum number of records to return.
- **offset** (*number*) – The offset from which to start to include records.

Returns Array – An array of Request objects that match the provided filters.

`fulfillment.Fulfillment.listTierConfigRequests (filters, orderBy, limit=100, offset=0)`

Returns a list of at most **limit** TierConfigurationRequest objects that match the provided filters. If no filter is passed, a page of Request in 'pending' status is returned.

See the *RequestService.list method* for further information about the filters object.

Arguments

- **filters** (*Object*) – A filter object.
- **orderBy** (*string/Array*) – The ordering criteria.
- **limit** (*number*) – The maximum number of records to return.
- **offset** (*number*) – The offset from which to start to include records.

Returns Array – An array of TierConfigurationRequest objects that match the provided filters.

`fulfillment.Fulfillment.updateRequest (id, request)`

Updates a Request object. Only a partial update can be performed on a Request object: developers can update only the **note** attribute of the Request and/or the asset parameters **value** or **value_error** attributes.

Arguments

- **id** (*string*) – The unique identifier of the Request object.
- **request** (*Object*) – The body of the request.

Returns Object – The updated Request object.

Examples:

```

{
  note: 'Test Note',
  asset: {
    params: [
      {
        id: 'param_a', // id is required
        value: 'value_of_param_a',
        value_error: 'This address is already used. Try another.'
      }
    ]
  }
}
```

`fulfillment.Fulfillment.updateRequestParameters (id, params, note)`

Updates the asset parameters of a Request object.

Arguments

- **id** (*string*) – The unique identifier of the Request object.

- **params** (*Array*) – Array of *Parameter* objects to update.
- **note** (*string*) – An optional **note** for the Request.

Returns Object – The updated Request object.

`fulfillment.Fulfillment.updateTierConfigRequest (id, request)`

Updates a *TierConfigurationRequest* object. Only a partial update can be performed on a *TierConfigurationRequest* object: developers can update only the **notew** attribute of the *TierConfigurationRequest* and/or the config parameters **value** or **value_error** attributes.

Arguments

- **id** (*string*) – The unique identifier of the *TierConfigurationRequest* object.
- **request** (*Object*) – The body of the request.

Returns Object – The updated *TierConfigurationRequest* object.

Examples:

```
{
  notes: 'Test Note',
  params: [
    {
      id: 'param_a', // id is required
      value: 'value_of_param_a',
      value_error: 'This address is already used. Try another.'
    }
  ]
}
```

`fulfillment.Fulfillment.updateTierConfigRequestParameters (id, params, notes)`

Updates the parameters of a *TierConfigurationRequest* object.

Arguments

- **id** (*string*) – The unique identifier of the *TierConfigurationRequest* object.
- **params** (*Array*) – Array of *Parameter* objects to update.
- **notes** (*string*) – An optional **notes** for the *TierConfigurationRequest*.

Returns Object – The updated Request object.

2.2.3 Services

class `accounts.AccountService ()`

The *AccountService* class provides methods to access the accounts endpoint of the Cloud Blue Connect API.

`accounts.AccountService.list ()`

Get a list of accounts based on the api key ownership.

Returns Array – Returns an Array of the *Account* object.

class `products.ProductService ()`

The *ProductService* class provides methods to access the products endpoint of the Cloud Blue Connect API.

`products.ProductService.getAssetParametersForFulfillmentByProduct (id)`

Retrieves the list of parameters of scope asset and phase fulfillment configured for the product.

Arguments

- **id** (*string*) – The unique product identifier for which retrieve the parameters.

Returns Array – An array with the Parameter objects.

`products.ProductService.getParametersByProduct (id)`
Retrieves the list of parameters configured for the product.

Arguments

- `id (string)` – The unique product identifier for which retrieve the parameters.

Returns Array – An array with the Parameter objects.

`products.ProductService.getProductAssetTemplates (id)`
Returns the list of templates configured for a product with scope “asset”.

Arguments

- `id (string)` – The unique identifier of the product.

Returns Array – An array of Template objects.

`products.ProductService.getProductTemplates (id)`
Returns the list of templates configured for a product.

Arguments

- `id (string)` – The unique identifier of the product.

Returns Array – An array of Template objects.

`products.ProductService.list ()`
Get a list of the latest version of products that has been published.

Returns Array – An Array of the Product object.

class `requests.RequestService ()`

The `RequestService` class provides methods to access the `Request` endpoint of the Cloud Blue Connect API.

`requests.RequestService.approve (id, request)`

Change the status of a `Request` to approved. To change the status of the `Request` to approved you must provide either a template id or a template tile.

Arguments

- `id (string)` – The unique identifier of the Request object.
- `request (Object)` – The request body.

Returns Object – The updated Request object.

Examples:

```
// request body using a template id
{
  template_id: 'TL-827-840-476'
}
```

```
// request body using an activation tile
{
  activation_tile: '<rendered text>'
}
```

`requests.RequestService.create (request)`
Create a new Request object.

Arguments

- **request** (*Object*) – The Request object to create.

Returns Object – A ‘partial’ Request object.

`requests.RequestService.fail(id, reason)`

Change the status of a Request to fail.

Arguments

- **id** (*string*) – The unique identifier of the Request object.
- **reason** (*string*) – The reason for which the Request has been failed.

Returns Object – The updated Request object.

`requests.RequestService.inquire(id, request)`

Change the status of a Request to inquire. To change the status of the Request to inquire you must provide either a template id or a template tile.

Arguments

- **id** (*string*) – The unique identifier of the Request object.
- **request** (*Object*) – The request body.

Returns Object – The updated Request object.

Examples:

```
// request body using a template id
{
  template_id: 'TL-827-840-476'
}
```

```
// request body using an activation tile
{
  activation_tile: '<rendered text>'
}
```

`requests.RequestService.list(filter, orderBy, limit=100, offset=0)`

Get a list of requests that match all the filter values provided as input. Some fields accept an array of values (search with the *in* operator).

Arguments

- **filter** (*Object*) – A filter object as described below.
- **filter.type** (*string/Array*) – The request type (purchase/change/suspend/resume/renew/cancel).
- **filter.status** (*string/Array*) – The status of the request (pending/inquiring/failed/approved).
- **filter.id** (*string/Array*) – The id of the request.
- **filter.assetId** (*string/Array*) – The id of the asset that belongs to the request.
- **filter.assetProductId** (*string/Array*) – The id of the product that belongs to asset of the request.
- **filter.assetProductName** (*string/Array*) – The name of the product that belongs to asset of the request (exact match case-insensitive).
- **filter.assetConnectionType** (*string/Array*) – The type of the connection.

- **filter.assetConnectionHubId** (*string/Array*) – The id of the hub involved in the asset connection.
- **filter.assetConnectionHubName** (*string/Array*) – The name of the hub involved in the asset connection.
- **filter.assetConnectionProviderId** (*string/Array*) – The id of the provider involved in the asset connection.
- **filter.assetConnectionProviderName** (*string/Array*) – The name of the provider involved in the asset connection.
- **filter.assetConnectionVendorId** (*string/Array*) – The id of the vendor involved in the asset connection.
- **filter.assetConnectionVendorName** (*string/Array*) – The name of the vendor involved in the asset connection.
- **filter.assetTiersCustomerId** (*string*) – The id of the customer account for this asset.
- **filter.assetTiersTier1Id** (*string*) – The id of T1 account for this asset.
- **orderBy** (*string/Array*) – The ordering criteria.
- **limit** (*number*) – The maximum number of records to return.
- **offset** (*number*) – The offset from which to start to include records.

Returns Array – An array of Request objects that match the provided filters.

`requests.RequestService.update(id, request)`

Update a Request object.

Arguments

- **id** (*string*) – The unique identifier of the Request object.
- **request** (*Object*) – The request body.

Returns Object – The updated Request object.

class `conversations.ConversationService()`

The ConversationService class provides methods to access the conversations endpoint of the Cloud Blue Connect API.

`conversations.ConversationService.createMessage(id, msg)`

Post a message to a Conversation object.

Arguments

- **id** (*string*) – The unique identifier of the Conversation object.
- **msg** (*string*) – The text of the message to post.

Returns Object – The newly created Message object.

`conversations.ConversationService.getConversation(id)`

Retrieves a Conversation object by id.

Arguments

- **id** (*string*) – The unique identifier of the Conversation object.

Returns Object – The Conversation object identified by id.

`conversations.ConversationService.getConversationsByObjectId(id)`

Returns a list of conversations attached to the business object specified by its id.

Arguments

- **id** (*string*) – The unique id of the business object attached to a conversation.

Returns Array – A list of conversations attached to a business object.

class `tierConfigRequests.TierConfigRequestService()`

The `TierConfigRequestService` class provides methods to access the `TierConfigRequest` endpoint of the Cloud Blue Connect API.

`tierConfigRequests.TierConfigRequestService.approve(id, request)`

Change the status of a `TierConfigurationRequest` to approved. To change the status of the `TierConfigurationRequest` to approved you must provide a `Template` id.

Arguments

- **id** (*string*) – The unique identifier of the `TierConfigurationRequest` object.
- **request** (*Object*) – The request body.

Returns Object – The rendered `Template`.

Examples:

```
// request body using a template id
{
  template: {
    id: 'TL-000-000-000'
  }
}
```

`tierConfigRequests.TierConfigRequestService.create(request)`

Create a new `TierConfigurationRequest` object.

Arguments

- **request** (*Object*) – The `TierConfigurationRequest` object to create.

Returns Object – A ‘partial’ Request object.

`tierConfigRequests.TierConfigRequestService.fail(id, reason)`

Changes the status of a `TierConfigurationRequest` to fail.

Arguments

- **id** (*string*) – The unique identifier of the `TierConfigurationRequest` object.
- **reason** (*string*) – The reason for which the `TierConfigurationRequest` has been failed.

`tierConfigRequests.TierConfigRequestService.inquire(id)`

Changes the status of a `TierConfigurationRequest` to inquire.

Arguments

- **id** (*string*) – The unique identifier of the `TierConfigurationRequest` object.

`tierConfigRequests.TierConfigRequestService.list(filter, orderBy, limit=100, offset=0)`

Get a list of tier configuration requests that match all the filter values provided as input. Some fields accept an array of values (search with the *in* operator).

Arguments

- **filter** (*string/Query*) – A filter object as described below.
- **filter.type** (*string/Array*) – The request type (setup/update).
- **filter.status** (*string/Array*) – The status of the request (tiers_setup/pending/inquiring/approved/failed).
- **filter.id** (*string/Array*) – The id of the tier configuration request.
- **filter.configurationId** (*string/Array*) – The id of the tier configuration object that belongs to the request.
- **filter.configurationTierLevel** (*string/Array*) – Tier level for product from customer perspective (1|2).
- **filter.configurationAccountId** (*string/Array*) – The id of the tier account of the tier configuration that belongs to the request.
- **filter.configurationProductId** (*string/Array*) – The id of the product of the tier configuration that belongs to the request.
- **filter.configurationAccountExternalUID** (*string/Array*) – The id of the user assignee of this tier configuration request.
- **filter.assigneeId** (*string*) – The id of the user assignee of this tier configuration request.
- **filter.unassigned** (*boolean*) – True to filter for unassigned requests, false otherwise.
- **orderBy** (*string/Array*) – The ordering criteria.
- **limit** (*number*) – The maximum number of records to return.
- **offset** (*number*) – The offset from which to start to include records.

Returns Array – An array of Request objects that match the provided filters.

`tierConfigRequests.TierConfigRequestService.pending(id)`

Changes the status of a TierConfigurationRequest to pending.

Arguments

- **id** (*string*) – The unique identifier of the TierConfigurationRequest object.

`tierConfigRequests.TierConfigRequestService.update(id, request)`

Update a TierConfigurationRequest object.

Arguments

- **id** (*string*) – The unique identifier of the TierConfigurationRequest object.
- **request** (*Object*) – The request body.

Returns Object – The updated TierConfigurationRequest object.

class `productActions.ActionService()`

The ActionService class provides methods to access the Action endpoint of the Cloud Blue Connect API.

`productActions.ActionService.get(id, productId, productVersion)`

Retrieve an Action object by its id.

Arguments

- **id** (*string*) – The unique identifier of the Action object.
- **productId** (*string*) – The unique identifier this Action object belongs to.

- **productVersion** (*string*) – The optional product’s version identifier.

Returns Object – The Action object identified by its id.

`productActions.ActionService.link(id, productId, productVersion)`

Retrieve an `ActionLink` object by the `Action` id.

Arguments

- **id** (*string*) – The unique identifier of the Action object.
- **productId** (*string*) – The unique identifier this Action object belongs to.
- **productVersion** (*string*) – The optional product’s version identifier.

Returns Object – The `ActionLink` object identified by the `Action` id.

`productActions.ActionService.list(productId, productVersion, limit=100, offset=0)`

Get a list of actions related to a product and optionally version of the product.

Arguments

- **productId** (*string*) – The unique identifier of the product these actions belongs to.
- **productVersion** (*string*) – The optional product’s version identifier.
- **limit** (*number*) – The maximum number of records to return.
- **offset** (*number*) – The offset from which to start to include records.

Returns Array – An array of `Action` objects.

2.2.4 Errors

class `HttpError` (*status, message*)

`HttpError` wraps any http response which status is `!= 2xx`.

Create a new instance of the `HttpError` class.

Arguments

- **status** (*number*) – the http status code.
- **message** (*string*) – an error message.

`HttpError.status`

type: `number`

The http status code.

class `APIError` (*status, message*)

`APIError` wraps any http response which status is `!= 2xx`.

Provides a quick and easy access to the JSON error.

The Cloud Blue Connect JSON error looks like:

Examples:

```
{
  "error_code": "SYS_001",
  "errors": [
    "error message 1"
  ]
}
```

`APIError`. **errorCode**

type: string

The error code returned by the server.

`APIError`. **errors**

type: Array

The array of error messages.

`APIError`. **json**

type: Object

The JSON error object.

INDICES AND TABLES

- genindex
- modindex
- search

**CHAPTER
FOUR**

LICENSE

© 2019 Ingram Micro Inc. All right reserved.

The Connect Javascript SDK is licensed under the [Apache License 2.0](#).

INDEX

A

accounts.AccountService() (class), 9
accounts.AccountService.list() (accounts.AccountService method), 9
APIError() (class), 15
APIError.errorCode (APIError attribute), 15
APIError.errors (APIError attribute), 16
APIError.json (APIError attribute), 16

C

ConnectClient() (class), 4
ConnectClient.accounts (ConnectClient attribute), 4
ConnectClient.actions (ConnectClient attribute), 4
ConnectClient.conversations (ConnectClient attribute), 4
ConnectClient.hubs (ConnectClient attribute), 4
ConnectClient.products (ConnectClient attribute), 5
ConnectClient.requests (ConnectClient attribute), 5
ConnectClient.tierAccounts (ConnectClient attribute), 5
ConnectClient.tierConfigRequests (ConnectClient attribute), 5
conversations.ConversationService() (class), 12
conversations.ConversationService.createMessage() (conversations.ConversationService method), 12
conversations.ConversationService.getConversation() (conversations.ConversationService method), 12
conversations.ConversationService.getConversationsByObjectId() (conversations.ConversationService method), 12

F

fulfillment.Fulfillment() (class), 5
fulfillment.Fulfillment.approveRequest() (fulfillment.Fulfillment method), 5

fulfillment.Fulfillment.approveRequestWithTemplate (fulfillment.Fulfillment method), 5
fulfillment.Fulfillment.approveTierConfigRequest() (fulfillment.Fulfillment method), 6
fulfillment.Fulfillment.approveTierConfigRequestWithTemplate (fulfillment.Fulfillment method), 6
fulfillment.Fulfillment.createRequest() (fulfillment.Fulfillment method), 6
fulfillment.Fulfillment.createTierConfigRequest() (fulfillment.Fulfillment method), 6
fulfillment.Fulfillment.failRequest() (fulfillment.Fulfillment method), 6
fulfillment.Fulfillment.failTierConfigRequest() (fulfillment.Fulfillment method), 6
fulfillment.Fulfillment.inquireRequest() (fulfillment.Fulfillment method), 7
fulfillment.Fulfillment.inquireRequestWithTemplate (fulfillment.Fulfillment method), 7
fulfillment.Fulfillment.listRequests() (fulfillment.Fulfillment method), 7
fulfillment.Fulfillment.listTierConfigRequests() (fulfillment.Fulfillment method), 8
fulfillment.Fulfillment.updateRequest() (fulfillment.Fulfillment method), 8
fulfillment.Fulfillment.updateRequestParameters() (fulfillment.Fulfillment method), 8
fulfillment.Fulfillment.updateTierConfigRequest() (fulfillment.Fulfillment method), 9
fulfillment.Fulfillment.updateTierConfigRequestParameters() (fulfillment.Fulfillment method), 9

H

HttpError() (class), 15
HttpError.status (HttpError attribute), 15

P

productActions.ActionService() (class), 14
productActions.ActionService.get() (productActions.ActionService method), 14
productActions.ActionService.link() (productActions.ActionService method), 15

productActions.ActionService.list ()
 (*productActions.ActionService method*), 15
products.ProductService () (*class*), 9
products.ProductService.getAssetParametersForFulfillmentByProduct ()
 (*products.ProductService method*), 9
products.ProductService.getParametersByProduct ()
 (*products.ProductService method*), 10
products.ProductService.getProductAssetTemplates ()
 (*products.ProductService method*), 10
products.ProductService.getProductTemplates ()
 (*products.ProductService method*), 10
products.ProductService.list () (*prod-
ucts.ProductService method*), 10

R

requests.RequestService () (*class*), 10
requests.RequestService.approve () (*re-
quests.RequestService method*), 10
requests.RequestService.create () (*re-
quests.RequestService method*), 10
requests.RequestService.fail () (*re-
quests.RequestService method*), 11
requests.RequestService.inquire () (*re-
quests.RequestService method*), 11
requests.RequestService.list () (*re-
quests.RequestService method*), 11
requests.RequestService.update () (*re-
quests.RequestService method*), 12

T

tierConfigRequests.TierConfigRequestService ()
 (*class*), 13
tierConfigRequests.TierConfigRequestService.approve ()
 (*tierConfigRequests.TierConfigRequestService
method*), 13
tierConfigRequests.TierConfigRequestService.create ()
 (*tierConfigRequests.TierConfigRequestService
method*), 13
tierConfigRequests.TierConfigRequestService.fail ()
 (*tierConfigRequests.TierConfigRequestService
method*), 13
tierConfigRequests.TierConfigRequestService.inquire ()
 (*tierConfigRequests.TierConfigRequestService
method*), 13
tierConfigRequests.TierConfigRequestService.list ()
 (*tierConfigRequests.TierConfigRequestService
method*), 13
tierConfigRequests.TierConfigRequestService.pending ()
 (*tierConfigRequests.TierConfigRequestService
method*), 14
tierConfigRequests.TierConfigRequestService.update ()
 (*tierConfigRequests.TierConfigRequestService
method*), 14