



# Moderation API

# shopreviews

Klantenvertellen & Kiyoh

## **Introduction**

The moderation API is a REST interface that is used to provide review data and shopreview moderation features. A consuming system will show this data in their application. The review data shown by the consuming system consists of two parts:

1. Statistics data
2. Review data

This document describes the functionality on a technical level, such that systems can connect to this interface.

## **Process Flow**

Some use cases are defined for the moderation API:

1. Get statistics data
2. Get all shopreviews
3. Respond to the review
4. Send the reviewer a request to change the review
5. Do a moderate request

## **Security**

The security for the moderation API will be token based. This means an authentication token needs to be provided as a header value in the HTTP request. For the moderation api this token is: X-Publication-Api-Token.

## **Error Handling**

An error message will occur, an HTTP error code will be returned plus an error message. For error messages the return format that is maintained is:

Data Element	Type	Description
httpCode	String	The http code for the error request
errorCode	String	The error code for this request, for example INTERNAL_SERVER_ERROR.
id	String	The id of the error request.
detailedError	String	Zero, one, or many detailed specifications of the error.
• errorCode	String	The error code specifying a more detailed error.
• message	String	A textual description of the error message.

## Latest updated locations and statistics

URL: <https://<baseurl>/v1/publication/review/locations/latest>

This API will get the locations statistics that have reviews added since a certain date or updated since a certain date. This operation is categorized as an HTTP GET operation.

URL Parameter	Mandatory	Type	Description
dateSince	Yes (or updatedSince)	String	DateTime of locations which have reviews created after given DateTime Supports yyyy-MM-dd'T'HH:mm:ss.SSSZ or yyyy-MM-dd
updatedSince	Yes (or dateSince)	String	DateTime of locations which have reviews updated after given DateTime Supports yyyy-MM-dd'T'HH:mm:ss.SSSZ or yyyy-MM-dd
start	No	Long	The position of the first result to retrieve ( e.g. 1st page start = 0, 2nd page start = 0+limit, nth page = n x limit )
limit	No	Integer	The number of rows that need to be retrieve, If not provided all records are fetched

## Get all reviews

URL: [https://<baseurl>/v1/publication/review/external?locationId=replace\\_locationId](https://<baseurl>/v1/publication/review/external?locationId=replace_locationId)

This operation will get the reviews for external reviews. This operation is categorized as an HTTP GET operation.

## Latest removed review

URL:

[https://<baseurl>/v1/publication/review/removed/latest?updatedSince=replace\\_updatedSince\\_frequent](https://<baseurl>/v1/publication/review/removed/latest?updatedSince=replace_updatedSince_frequent)

This operation will get the reviews that have been removed since a certain date. This operation is categorized as an HTTP GET operation.

Review platform	baseurl
Kiyoh	www.kiyoh.com
Klantenvertellen	www.klantenvertellen.nl

Review platform	tenantId
Kiyoh	98
Klantenvertellen	99

## Review change request

URL: <https://<baseurl>/v1/publication/review/changerequest>

This operation will submit a change request to reviewer to update the shopreview. This operation is categorized as an HTTP PUT operation. Input body:

```
{  
  "locationId": "replace_locationId",  
  "tenantId": "replace_tenantId",  
  "reviewId": "replace_reviewId"  
}
```

All fields above are mandatory.

## Review abuse request

URL : <https://<baseurl>/v1/publication/review/abuse>

This operation will submit a change request to reviewer to update the review. This operation is categorized as an HTTP PUT operation. Input body:

```
{  
  "locationId": "replace_locationId",  
  "tenantId": "replace_tenantId",  
  "reviewId": "replace_reviewId",  
  "abuseReason": "replace_reason_abuse"  
}
```

All fields above are mandatory.

## Review Response

URL : <https://<baseurl>/v1/publication/review/response>

This operation will submit a response to the review. This operation is categorized as an HTTP PUT operation. Input body:

```
{  
  "locationId": "replace_locationId",  
  "tenantId": "replace_tenantId",  
  "reviewId": "replace_reviewId",  
  "response": "replace_response",  
  "reviewResponseType": "replace_PRIVATE_OR_PUBLIC",  
  "responseEmail": "replace_true_or_false"  
}
```

All fields above are mandatory.